PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-044150

(43)Date of publication of application : 08.02.2002

(51)Int.Cl. H04L 12/56

(21)Application number : 2001-154076

(22)Date of filing : 23.05.2001

(71)Applicant : ALCATEL INTERNETWORKING (PE) INC

(72)Inventor : TALLEGAS MATHIEU
FLOM KERRY
DENIS PAUL

(30)Priority

Priority number :	Priority date :	Priority country :
2000 206617	24.05.2000	US
2000 206996	24.05.2000	US
2000 220335	24.07.2000	US

(54) PACKET PROCESSOR WITH MULTI-LEVEL POLICING LOGIC

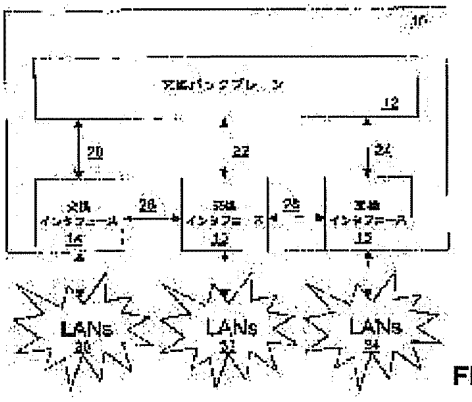

FIG. 1

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a switch including a back plane and packet processors.

SOLUTION: One or more packet processors include a multi-level policing logic. The packet processor receives a packet and classifiers the packet into policing available groups. The packet is compared with a bandwidth contact defined as to the policing available group. A policing database is used to apply nest retrieval to the packet to identify groups and to retrieve policing data with respect to the policing available groups. This policing result it combined into one policing result by adopting the policing result in the worst case, this is applied as recommendations to a disposition logic and the disposition is decided to the packet in combination with other disposition recommendations.

CLAIMS

[Claim(s)]

[Claim 1]A packet switching controller which it has an input part which receives a packet, and a regulation element which classifies a packet into two or more policing available groups, and a packet is compared with one or more bandwidth contracts that a policing available group was defined, and generates one or more policing results.

[Claim 2]In order that a regulation element may search the 1st regulated data and the 2nd policing available group identifier including a regulation database, the 1st policing available group identifier is applied to a regulation database, The packet switching controller according to claim 1 with which the 1st regulated data is applied in order to generate the 1st policing result, the 2nd policing available group identifier is applied to a regulation database in order to search the 2nd regulated data, and the 2nd regulated data is applied in order to generate the 2nd policing result.

[Claim 3]The packet switching controller according to claim 1 with which it has further an arrangement engine which makes an arrangement decision to a packet, and an

arrangement engine uses a policing result and other at least one arrangement advice, and a policing result makes an arrangement decision to a packet including one or more arrangement advice.

[Claim 4]The packet switching controller according to claim 1 which combines a policing result with one result by taking in a policing result in a case of being the worst.

[Claim 5]A method characterized by comprising the following of processing a packet using a regulation element.

A step which receives a packet.

A step which classifies a packet into two or more policing available groups.

A step which generates one or more policing results as compared with one or more bandwidth contracts of having defined a packet about a policing available group.

[Claim 6]A regulation element applies the 1st policing available group identifier to a regulation database including a regulation database, The 1st regulated data and a step which searches the 2nd policing available group identifier, Use the 1st regulated data and a step which generates the 1st policing result, and the 2nd policing available group identifier are applied to a regulation database, A method of containing further a step which searches the 2nd regulated data, and a step which uses the 2nd regulated data and generates the 2nd policing result of processing the packet according to claim 5.

[Claim 7]A method of processing the packet according to claim 5 that a policing result uses a policing result and other at least one arrangement advice, and is further provided with a step which makes an arrangement decision to a packet including one or more arrangement advice.

[Claim 8]A method of containing further a step which combines a policing result with one result by taking in a policing result in a case of being the worst of processing the packet according to claim 5.

[Claim 9]A method of regulating a data packet which a data communication switch received, comprising:

A step which classifies a data packet into two or more policing available groups.

A step which identifies regulated data related with one or more policing available groups.

A step which generates one or more policing results which apply regulated data and receive a policing available group.

A step which advises arrangement of a data packet from a policing result.

[Claim 10]A way according to claim 9 a specific policing available group identifies a type of application to regulate.

[Claim 11]A way according to claim 9 regulated data includes information about bandwidth restrictions specified about at least one policing available group.

[Claim 12]A way according to claim 9 a policing result shows whether a data packet should be transmitted.

[Claim 13]A method according to claim 9 of showing whether a policing result is proper although a data packet drops.

[Claim 14]A method according to claim 9 of showing whether a policing result should drop a data packet.

[Claim 15]A way according to claim 9 a step which advises arrangement contains a step which creates advice combining a policing result.

[Claim 16]A method according to claim 9 including that a step which advises arrangement chooses one of the policing results as advised arrangement.

[Claim 17]A method according to claim 9 of containing further a step which updates regulated data based on advised arrangement.

[Claim 18]A method of regulating a data packet which a data communication switch received, comprising:

A step which creates a regulation database including two or more regulation data entries which specify regulated data to two or more policing available groups.

The 1st regulated data that applies the 1st identifier and is related with the 1st policing available group.

A step which searches the 2nd identifier that identifies the 2nd policing available group.

A step which applies the 1st regulated data and generates the 1st policing result, a step which applies the 2nd identifier and searches the 2nd regulated data, a step which applies the 2nd regulated data and generates the 2nd policing result, and a step which advises arrangement of a data packet from the 1st and 2nd policing results.

[Claim 19]A way according to claim 18 a specific policing available group identifies a type of application to regulate.

[Claim 20]A way according to claim 18 regulated data includes information about bandwidth restrictions specified about a policing available group.

[Claim 21]A way according to claim 18 a policing result shows whether a data packet should be transmitted.

[Claim 22]A method according to claim 18 of showing whether a policing result is proper although a data packet drops.

[Claim 23]A method according to claim 18 of showing whether a policing result should drop a data packet.

[Claim 24]A way according to claim 18 a step which advises arrangement creates advice combining the 1st and 2nd policing results.

[Claim 25]A method according to claim 18 including that a step which advises arrangement chooses one of the 1st or 2nd policing result as advised arrangement

further.

[Claim 26]A method according to claim 18 of containing further a step which updates the 1st or 2nd regulated data based on advised arrangement.

[Claim 27]A regulation engine for a data communication node with which a regulation engine classifies a packet into two or more policing available groups, and generates each of a policing result about each of a policing available group as compared with each of a bandwidth contract of a packet.

[Claim 28]In order to search the 1st regulated data and the 2nd policing available group identifier, the 1st policing available group identifier is applied to a regulation database, A regulation engine for a data communication node with which the 1st regulated data is applied in order to generate the 1st policing result, the 2nd policing available group identifier is applied to a regulation database in order to search the 2nd regulated data, and the 2nd regulated data is applied in order to generate the 2nd policing result.

[Claim 29]A packet processor which it has an input part which receives a packet, and a control means which classifies a packet into two or more policing available groups, and a packet is compared with one or more bandwidth contracts that a policing available group was defined, and generates one or more policing results.

[Claim 30]In order that a control means may search the 1st regulated data and the 2nd policing available group identifier including a regulation database, the 1st policing available group identifier is applied to a regulation database, The packet processor according to claim 29 to which the 1st regulated data is applied in order to generate the 1st policing result, the 2nd policing available group identifier is applied to a regulation database in order to search the 2nd regulated data, and the 2nd regulated data is applied in order to generate the 2nd policing result.

[Claim 31]The packet processor according to claim 29 which is further equipped with an arrangement means which makes an arrangement decision to a packet and to which an arrangement means makes an arrangement decision to a packet using a policing result and other at least one arrangement advice including arrangement advice of one or more [policing result].

[Claim 32]The packet processor according to claim 29 which combines a policing result with one result by taking in a policing result in a case of being the worst.

[Claim 33]A packet switching controller is further provided with a debiting element, The packet switching controller according to claim 1 which has a related token bucket which at least one bandwidth contract shows available bandwidth under said bandwidth contract, and judges DEBITTO [element / a debiting element uses a policing result and / a related token bucket].

[Claim 34]A packet switching controller is further provided with a debiting element, It

has a related token bucket in which bandwidth with at least one bandwidth contract available under a bandwidth contract is shown, Until it provides with arrangement determination a debiting element used in order to judge DEBITTO [an arrangement engine / a related token bucket], The packet switching controller according to claim 3 which keeps unchanged DEBITTO [a debiting element / a packet size / a related token bucket].

[Claim 35]A method of having a related token bucket in which bandwidth with at least one bandwidth contract available under a bandwidth contract is shown, and including further judging DEBITTO [a related token bucket] using a policing result of processing the packet according to claim 5.

[Claim 36]A method of having a related token bucket in which bandwidth with at least one bandwidth contract available under a bandwidth contract is shown, and including further judging DEBITTO [a packet size / a related token bucket] using arrangement determination of processing the packet according to claim 7.

[Claim 37]Arrangement advice from a policing result and other at least one arrangement advice are used, A method of regulating the data packet according to claim 11 which contains further a step which generates arrangement determination to a data packet, and a step which judges whether information about bandwidth restrictions is updated using arrangement determination.

[Claim 38]Arrangement advice from the 1st and 2nd policing results and other at least one arrangement advice are used, A method of regulating the data packet according to claim 20 which contains further a step which generates arrangement determination to a data packet, and a step which judges whether information about bandwidth restrictions is updated using arrangement determination.

[Claim 39]The regulation engine according to claim 27 which judges whether available bandwidth is updated based on a policing result under a bandwidth contract.

[Claim 40]A packet processor is further provided with a DEBITTO means, and it at least one bandwidth contract, Until it provides with arrangement determination a debiting means to use it in order to judge DEBITTO [it has a related token bucket in which available bandwidth is shown under a bandwidth contract, and / an arrangement means / a related token bucket], The packet processor according to claim 31 which keeps unchanged DEBITTO [a debiting means / a packet size / a related token bucket].

[Claim 41]A data regulation method comprising:

A step which receives a packet.

A step which adds a time credit to the 1st token count, and generates the 2nd token count.

A step which applies the 2nd token count and generates a policing result to a packet.

A step which judges whether a policing result is applied, a SAIZUDE bit is subtracted from the 2nd token count, and the 3rd token count is generated, A step which subtracts a SAIZUDE bit from the 2nd token count, and generates the 3rd token count when subtraction applies a policing result and it is judged.

[Claim 42]A step which receives the 2nd packet, and a step which adds the 2nd time credit to the 2nd token count, and generates the 4th token count when the 3rd token count is not generated, A step which adds the 2nd time credit to the 3rd token count, and generates the 4th token count when the 3rd token count is generated, A data regulation method according to claim 41 which applies the 4th token count and contains further a step which generates a policing result to the 2nd packet.

[Claim 43]A data regulation method comprising:

A step which receives a packet.

A step which adds a time credit to the 1st token count, and generates the 2nd token count.

A step which generates a policing result to a packet with the application of the 2nd token count.

A step which generates an arranging result to a packet with the application of a policing result, A step which judges whether an arranging result is applied, a SAIZUDE bit is subtracted from the 2nd token count, and the 3rd token count is generated, A step which subtracts a SAIZUDE bit from the 2nd token count, and generates the 3rd token count when subtraction applies an arranging result and it is judged.

[Claim 44]A data regulation method according to claim 43 which applies a policing result as advice which has other at least one advice, and generates an arranging result.

[Claim 45]A data regulation method comprising:

A step which receives a packet.

A step which adds a time credit to each of a token count, and generates each of the 2nd token count.

A step which generates a policing result to a packet with the application of each of the 2nd token count.

A step which judges whether a policing result is applied, a SAIZUDE bit is subtracted from at least one of the 2nd token counts, and at least one 3rd token count is generated, A step which subtracts a SAIZUDE bit from at least one of the 2nd token counts, and generates at least one 3rd token count when subtraction applies a policing result and it is judged.

[Claim 46]A data regulation method comprising:

A step which receives a packet.

A step which adds a time credit to each of a token count, and generates each of the 2nd

token count.
A step which generates a policing result to a packet with the application of each of the 2nd token count.
A step which generates an arranging result to a packet with the application of a policing result, A SAIZUDE bit is subtracted from at least one of the 2nd token counts with the application of an arranging result, A step which judges whether at least one 3rd token count is generated, and a step which subtracts a SAIZUDE bit from at least one of the 2nd token counts, and generates at least one 3rd token count when subtraction applies an arranging result and it is judged.

DETAILED DESCRIPTION

[Detailed Description of the Invention]
[0001]
[Field of the Invention]. Applied for cross-reference this application of related application on May 24, 2000. The U.S. provisional application 60th of the name "System and Method for Enhanced Line Cards" / No. 206,617, The U.S. provisional application 60th of the name "Flow Resolution Logic System and Method" for which it applied on May 24, 2000 / No. 206,996, And the U.S. provisional application 60th of a name / right of priority of No. 220,335 of "Programmable Packet Processor" for which it applied on July 24, 2000 is charged.
All these contents are thoroughly included in this specification by reference.
This application includes the theme related to the theme currently indicated by U.S. patent application 09th of the name "Programmable Packet Processor withFlow Resolution Logic" for which it applied on December 28, 2000 / No. 751,194. These contents are thoroughly included in this specification by reference.
[0002]Generally this invention relates to the data communication switch which uses speed regulation (rate policing) of the plural level to a data packet for details more about a data communication switch.
[0003]
[Description of the Prior Art]Since the customer with the qualification for receiving a different quality of service (QoS) is vying in the available bandwidth of the network resource which is a common set, speed regulation is set to the data communication network, and is becoming still more important. Usually, speed regulation classifies each packet into one policy group, and is attained in each switch by comparing the classified packet with one or more bandwidth contracts that the group was defined. It is possible to transmit a packet based on the identified bandwidth contract, to attach and transmit

the mark of an abandonment eligibility (discard eligible), or to discard.

[0004]

[Problem(s) to be Solved by the Invention]The existing speed regulation method usually regulates the traffic of data for every port regardless of other information about traffic. Usually, when congestion arises, the data exceeding the speed for which the customer applied is marked as it should be dropped. therefore -- a customer is usually based on the specific application relevant to data -- etc. -- based on the type of data, it does not have the pliability to drop a certain kind of data selectively.

[0005]Since the request of doubling a communication network with the demand which the customer individualized is becoming strong, pliability is increasing, but it is so desirable for operation to provide the regulation logic which is not complicated that line speed is reduced remarkably.

[0006]

[Means for Solving the Problem]According to one embodiment of this invention, a packet switching controller is provided. A packet switching controller contains an input part which receives a packet, and a regulation element which classifies a packet into two or more policing available groups. A packet is compared with one or more bandwidth contracts that a policing available group was defined, and generates one or more policing results.

[0007]According to other embodiments of this invention, a method of processing a packet is provided. A packet is received and it classifies into two or more policing available groups. One or more policing results are generated as compared with one or more bandwidth contracts of having defined a packet about a policing available group.

[0008]According to other embodiments of this invention, a method of regulating a data packet which a data communication switch received is provided. A data packet is classified into two or more policing available groups. Subsequently, regulated data related with one or more policing available groups is identified. Regulated data is applied, one or more policing results which receive a policing available group are generated, and arrangement (disposition) of a data packet is advised from a policing result.

[0009]According to other embodiments of this invention, a method of regulating a data packet which a data communication switch received is provided. A regulation database include two or more regulation data entries which specify regulated data to two or more policing available groups is built. The 1st identifier is applied and the 1st regulated data related with the 1st policing available group and the 2nd identifier that identifies the 2nd policing available group are searched. Subsequently, the 1st regulated data is applied and the 1st policing result is generated. The 2nd identifier is applied and the

2nd regulated data is searched. Subsequently, the 2nd regulated data is applied and the 2nd policing result is generated. Arrangement of a data packet is advised from the 1st and 2nd policing results.

[0010]According to other embodiments of this invention, a regulation engine for a data communication node is provided. A regulation engine classifies a packet into two or more policing available groups. As compared with each of a bandwidth contract of a packet, each policing result is generated about each of a policing available group.

[0011]According to other embodiments of this invention, a regulation engine for a data communication node is provided. In order to search the 1st regulated data and the 2nd policing available group identifier, the 1st policing available group identifier is applied to a regulation database. In order to generate the 1st policing result, the 1st regulated data is applied, and in order to search the 2nd regulated data, the 2nd policing available group identifier is applied to a regulation database. In order to generate the 2nd policing result, the 2nd regulated data is applied.

[0012]According to other embodiments of this invention, a packet processor is provided. A packet processor includes an input part which searches a packet, and a control means which classifies a packet into two or more policing available groups. A packet is compared with one or more bandwidth contracts that a policing available group was defined, and generates one or more policing results.

[0013]

[Embodiment of the Invention]I. The network environment containing the packet switching node 10 is shown by the schematic diagram 1. A packet switching node can be called a switch, a data communication node, or a data communication switch. Interconnection of the packet switching node 10 is carried out to LAN 30, 32, and 34, respectively, and it includes the exchange interfaces 14, 16, and 18 in which interconnection is mutually carried out by the data paths 20, 22, and 24 via the exchange back plane 12. As for the exchange back plane 12, it is preferred that an exchange fabric is included. An exchange interface can be combined with each other according to the control routes 26 and 28.

[0014]The exchange interfaces 14, 16, and 18 Media-access-control (MAC) bridging, Internet Protocol (IP) routing, etc., It is preferred to send a packet to each group of LAN 30, 32, and 34, and to send a packet from there according to one or more operational communications protocols. The switching node 10 is only shown for the purpose of illustration. Actually, a packet switching node exceeds three or can include less than three exchange interfaces.

[0015]Drawing 2 is a block diagram of the exchange interface 50 in one embodiment of this invention. The exchange interface 50 can suppose that it is the same as that of the

exchange interfaces 14, 16, and 18 of drawing 1, etc. The exchange interface 50 contains the access controller 54 combined between LAN and the packet switching controller 52. A medium access controller (MAC) can be included by the access controller 54, for example. It is preferred to carry out processing which receives the inbound packet which left LAN, carries out the physical layer and MAC layer operation for which it does not depend on a flow to an inbound packet, and transmits an inbound packet to the packet switching controller 52 and for which it depends on a flow. As for the access controller 54, it is preferred to receive an outbound packet from the packet switching controller 52, and to transmit a packet on LAN. Physical operation and MAC layer operation are carried out to an outbound packet, and the access controller 54 can be transmitted on LAN after that.

[0016]In order to cope with the packet which has large various communications protocols, the programmable thing of the packet switching controller 52 is preferred. As for the packet switching controller 52, it is preferred to receive an inbound packet, to classify a packet, to correct a packet according to flow information, and to transmit the changed packet on exchange back planes, such as the exchange back plane 12 of drawing 1. As for the packet switching controller 52, it is preferred to receive the packet corrected by other packet switching controllers via an exchange back plane, to transmit it to the access controller 54, and to advance on LAN. Exit processing (egress processing) is performed to what the packet chose, it transmits to the access controller 54 after that, and the packet switching controller 52 can be transmitted on LAN.

[0017]Drawing 3 is a block diagram of the programmable packet switching controller 100 in one embodiment of this invention. For example, the programmable packet switching controller 100 can suppose that it is the same as that of the packet switching controller 52 of drawing 2. As for the programmable packet switching controller 100, it is preferred to have the flow analysis logic which classifies and routes the ingress flow of a packet. As for a programmable packet switching controller, because of programmable character, it is preferred to provide the pliability coping with the protocol and/or the updating possibility of the field that many differ. A programmable packet switching controller can be called under the name of the others generally used by a packet switching controller, a switching controller, the program packet processor, the network processor, the communication processor, or the person skilled in the art.

[0018]The programmable packet switching controller 100 contains the packet buffer 102, the packet classification engine 104, the application engine 106, and the regulation engine 120. A regulation engine can also be called a regulation element. Few [that it is more or] constitution elements can be included by the packet switching controller of other embodiments. For example, the pattern-matching module which investigates

compatibility comparing a part of packet with a predetermined pattern can be included by the packet switching controller of other embodiments. The packet switching controller of other embodiments can edit an inbound packet, and the edit module which generates an outbound packet can be included.

[0019]As for the programmable packet switching controller 100, it is preferred to receive the inbound packet 108. Although an Ethernet (registered trademark) frame, an ATM cell, TCP/IP, and/or UDP/IP packet can be included by the packet, it is not limited to this. It is possible for the data unit of other layers 2 (a data link/MAC layer), the layer 3 (network layer), or the layer 4 (transport layer) to be included. For example, the packet buffer 102 can receive an inbound packet from one or more media-access-control (MAC) layer interfaces via Ethernet.

[0020]As for the packet which received, being stored in the packet buffer 102 is preferred. Packet FIFO which receives a packet and is stored temporarily can be included by the packet buffer 102. As for the packet buffer 102, it is preferred to provide the packet classification engine 104 and the application engine 106 with the stored packet or its part, and to process it.

[0021]The edit module which edits a packet and is carried forward outside from a switching controller as the outbound packet 118 after that can be included by the packet buffer 102. the edit program in which an edit module creates an edit program in real time -- building -- it is possible for an engine and/or the edit engine which corrects a packet to be included. The application engine 106 has a preferred thing which can include the arrangement determination of a packet and for which the packet buffer 102 is provided with the application data 116. an edit program -- building -- as for an engine, it is preferred to use application data and to create an edit program. The outbound packet 118 can be transmitted to communication networks, such as Ethernet, via an exchange fabric interface.

[0022]One of a header-data extractor and the header data caches or both can be included by the packet buffer 102. It is preferred to use a header-data extractor and to store in a header data cache the field which extracted and extracted one or more fields from the packet as extraction header data. Although a part or all of packet headers can be included by extraction header data, they are not limited to this. For example, a header data cache is able to store the first N byte of each frame in an Ethernet system.

[0023]As for extraction header data, it is preferred to provide and process to the packet classification engine 104 as the output signal 110. The application engine can pass the interface 114, and can require and receive extraction header data. Extraction header data The MAC Address of the layer 2, 802.1 P/Q tag status, Although it is possible for one or more of the sealing (encapsulation) type of the layer 2, the protocol type of the

layer 3, the address of the layer 3, a ToS (type of service) value, and the port number of the layer 4 to be included, it is not limited to this. At other embodiments, the inbound packet whole [other than it instead of the extracted header data] can be included by the output signal 110. In other embodiments, it is possible to use the packet classification engine 104, and to make extraction header data a format suitable for use of application engine, and/or to load data to a header data cache.

[0024]As for the packet classification engine 104, it is preferred that programmable microcode drive embedding type processing engine is included. As for the packet classification engine 104, being combined with command RAM (IRAM) (not shown) is preferred. As for packet classification engine, it is preferred to read and execute the command stored in IRAM. In one embodiment, many of commands which packet classification engine executes are condition jumps. According to this embodiment, classification logic contains the determination tree in which that the packet sorting of a different type is shown has a desirable leaf in an end part. As for the brunch of a determination tree, it is preferred to be chosen based on comparison with instruction conditions and the header field stored in the header data cache. At other embodiments, classification logic cannot be based on a determination tree.

[0025]As for the application engine 106, in one embodiment of this invention, it is preferred that two or more programmable sub engine has the pipeline architecture currently pipelined linearly. As for each programmable sub engine, it is preferred to carry out operation to a packet, it is a "bucket brigade" method and it is preferred to transmit a packet to the following programmable sub engine. the start signal 112 is used for packet classification engine -- the [of application engine] -- it is preferred by starting 1 programmable sub engine to start pipelining packet processing. The start signal 112 can include discernment of one or more programs executed with the application engine 106. The start signal 112 can include packet classification information. As for the programmable sub engine of application engine, it is preferred to have the direct access to header data and the extraction field stored in the header data cache via the interface 114.

[0026]As for a decision-making stage, although the application engine can include other processing stages which programmable sub engine does not perform, it is preferred to perform with programmable sub engine and to increase pliability. At other embodiments, the application engine can include other treatment structures.

[0027]As for the arrangement determination included in the application data 116, being provided for the regulation engine 120 is preferred. As for the regulation engine 120, it is preferred again to receive one or more regulation ID124. As for the regulation engine 120, it is preferred to use arrangement determination and regulation ID and to generate

one or more regulation advice 122. Regulation advice can be considered as the type of arrangement advice, and it is also possible to call it a policing result. It is preferred to generate the application data which the application engine 106 is provided with regulation advice, and it is used with other arrangement advice, and can include arrangement determination.

[0028]II. The programmable arrangement logic diagram 4 is a block diagram of the packet switching controller 130 which has programmable arrangement logic. The packet switching controller 130 can suppose that it is the same as that of the packet switching controller 100 of drawing 3, for example. A packet switching controller contains the packet buffer 132, the packet classification engine 134, the pattern-matching search logic 136, the application engine 138, and the regulation engine 166.

[0029]Application engine contains the sauce searching engine 140, the address searching engine 142, and the arrangement engine 144. As for packet classification engine, sauce searching engine, address searching engine, and an arrangement engine, it is preferred that it is programmable using one or more application programs. That is, as for the sub engine of packet classification engine and application engine, it is preferred respectively that programmable microcode drive embedding type processing engine is included. In other embodiments, it is possible to be hardware, namely, to carry one or more of these engines out as a hard-wired logic. The regulation engine 166 can be carried out with a hard-wired logic or programmable microcode drive embedding type processing engine.

[0030]As for the packet buffer 132, it is preferred to receive and store the inbound packet 146. An inbound packet or its thing [providing the packet classification engine 134 with 148 in part] of a packet buffer is preferred. As for packet classification engine, it is preferred to use the application program currently programmed on it and to classify a packet, and it is preferred to provide the application engine 138 with the program discernment 152. In details, it is more preferred to provide the sauce searching engine 140, the address searching engine 142, and the arrangement engine 144 of application engine with the program discernment 152. According to one embodiment of this invention, the packet classification engine 134 contains the classification logic based on a decision tree.

[0031]It is preferred to use the program discernment 152 and to choose the application program executed by each of sauce searching engine, address searching engine, and an arrangement engine. As for the application program executed with sauce searching engine, address searching engine, and an arrangement engine, it is preferred to be selectively chosen based on packet classification information at least. Packet

classification information can be provided with program discernment.

[0032]An inbound packet or its thing [providing the pattern-matching search logic 136 with 150 in part] of a packet buffer is preferred. As for pattern-matching search logic, it is preferred that the pattern defined for comparing a part of packet or packet in advance is included. For example, both some packet header data, some packet payload data, or packet header data and packet payload data [some of] can be included by a part of packet used for pattern matching. At other embodiments, the pattern defined in advance can be existed in the external memory which pattern-matching search logic accesses for pattern matching. At other embodiments, a matching pattern can be changed by the packet switching controller working.

[0033]After comparing, it is preferred to provide the application engine 138 with 154 as a result of comparison. In details, it is more preferred to provide the arrangement engine 144 of application engine with 154 as a result of comparison. In one embodiment, only when there is consistency, it is possible to provide an arrangement engine with a result.

[0034]As for the sauce searching engine 140, it is preferred by using the source address of an inbound packet and carrying out source address search selectively at least to generate the arrangement advice 160 to an inbound packet. As for the arrangement advice 160, it is preferred that it is dependent on the application program executed with the sauce searching engine 140 according to the program discernment provided with packet classification engine. As for the arrangement advice 160, it is preferred to include the security advice to an inbound packet.

[0035]In other embodiments, it is possible to use the sauce searching engine 140 and to build one or more keys, and it is possible to use this subsequently and to search an address table for the source addresses (IPSA etc.) to an inbound packet. Although virtual LAN discernment (VLAN ID), application discernment (APP ID), and one or more of IPSA can be included by the key, it is not limited to this. It is also possible to use one or more keys built with the sauce searching engine 140, for example, to decide upon arrangement advice of security advice etc.

[0036]As for the address searching engine 142, it is preferred to receive the output 156 from the sauce searching engine 140. The output 156 can include the result of the key used in order to search for a source address, and/or search. As for address searching engine, it is preferred to execute the application program identified with the packet classification engine 134, and to generate one or more regulation identifiers (ID) 168. Regulation ID168 can be selectively based on the destination address search which uses the destination address of an inbound packet at least.

[0037]As for the regulation engine 166, it is preferred to use regulation ID168 as a key

and to access the regulated data of a regulation data table. As for the regulation engine 166, it is preferred to use the accessed regulated data and to generate one or more regulation advice 170. When an arrangement engine uses regulation advice and other arrangement advice, it is preferred to generate the application data which can include arrangement determination. As for the pattern-matching result 154, when the pattern-matching search logic 136 finds consistency, it is preferred to give priority over regulation advice. It is preferred by using regulation advice and choosing the regulation advice in the case of being the worst to generate one advice. The regulation engine can also carry out an accounting (accounting) function.

[0038]In other embodiments, it is possible to use the address searching engine 142 and to build one or more keys, and it is possible to use this subsequently and to search the destination addresses (IPDA etc.) of an inbound packet in an address table. Although virtual LAN discernment (VLAN ID), application discernment (APP ID), and one or more of IPDA can be included by the key, it is not limited to this.

[0039]Although the arrangement engine 144 includes security advice of the arrangement advice 160, the regulation advice 170, and the pattern-matching result 154, it is preferred to receive some arrangement advice which is not limited to this. As for an arrangement engine, it is preferred to generate the arrangement determination 162 based on arrangement advice and packet sorting, and/or program discernment. One of the arrangement advice can be included by the arrangement determination 162. Generally, the pattern-matching result 154 can give priority over the regulation advice 170, and the regulation advice can give priority over security advice of the arrangement advice 160. Although one or more of account data, routing data, and regulated data can be included by the arrangement determination 162, they may be some application data which are not limited to this.

[0040]It is preferred to use it for edit of the inbound packet which provides a packet buffer with arrangement determination and is provided as the outbound packet 164. It is preferred to supply arrangement determination to a regulation engine again for regulation and accounting. For example, when an inbound packet is dropped, the regulation engine should recognize that. A regulation engine can be included by address searching engine at other embodiments. In such a case, as for arrangement determination, it is preferred to be provided for address searching engine for regulation and accounting.

[0041]Drawing 5 is a process-flow figure which uses two or more arrangement advice and classification information, and generates arrangement determination by a program. It is Step 180 and, as for packet buffers, such as the packet buffer 132 of drawing 4, it is preferred to receive an inbound packet, for example. In a packet buffer, it is possible to

extract packet header data and to store in a header data cache.

[0042]Header data can be included by a part of inbound packet or inbound packet, for example, it is preferred to be provided for pattern-matching search logic, such as the pattern-matching search logic 136 of drawing 4. It is preferred to generate pattern-matching advice at Step 182, as pattern-matching search logic carries out pattern-matching search between a part of inbound packet or inbound packet, and a predetermined pattern and is shown by Step 188. For example, a predetermined pattern can be contained in an internal memory or external memory. According to other embodiments, a matching pattern may change dynamically.

[0043]On the other hand, it is also preferred to provide packet classification engine, such as the packet classification engine 134 of drawing 4, with an inbound packet or its part, for example. It is preferred it to be preferred to classify a packet as for packet classification engine, and to identify an application program at Step 184, based on the classification of a packet. It is preferred to provide the sauce searching engine of application engine, such as the application engine 138 of drawing 4, address searching engine, and an arrangement engine with program discernment at Step 186, for example. As for program discernment, it is preferred that the application program executed with such sub engine is shown. It is preferred to provide sauce searching engine, address searching engine, and an arrangement engine with packet classification information. As for sauce searching engine, it is preferred to generate security advice at Step 190, and, as for a regulation engine, on the other hand, it is preferred to use regulation ID from address searching engine, and to generate regulation advice at Step 192.

[0044]It is preferred to provide an arrangement engine with pattern-matching advice, security advice, and regulation advice at Step 194. As for an arrangement engine, it is preferred to use one or more of the selected application program and arrangement advice, and to generate arrangement determination. It is preferred to provide a packet buffer with arrangement determination, to use this, to edit an inbound packet at Step 196, and to transmit as an outbound packet. It is preferred to supply arrangement determination to a regulation engine again at Step 198 for example, for regulation, accounting, etc.

[0045]III. As for a regulation engine, in one embodiment of multilevel regulation this invention, it is preferred to use the multilevel regulation logic which regulates the traffic which he follows through a packet switching controller based on two or more policy groups. As for a customer, in its bandwidth contract, it is preferred to specify bandwidth applicable to suitable policy groups and those groups. It is possible to specify that a customer pays 1 Gbps of data traffic about a specific port in his bandwidth contract in an illustration scenario. The customer can assign a different data flow limit

to the subnet of his company. For example, the customer can limit an engineering subnet to 300Mbps, and can limit an accounting subnet to 100Mbps. A customer is the whole company and can specify that it limits the traffic of a web to 200Mbps. Instead of regulating only traffic for every port regardless of the type of traffic, therefore, web traffic, It is possible to identify and regulate the traffic which makes an engineering subnet or an accounting subnet the source of dispatch based on each threshold.

[0046]It is also possible to judge QoS operation by the bandwidth contract between a service provider and a customer. The QoS operation can identify QoS applicable to the traffic which fulfills flow conditions. Maximum band width, minimum bandwidth, peak bandwidth, a priority, waiting time, a jitter, the maximum cue depth, the maximum queue buffer, etc. can be shown by QoS operation.

[0047]As a part of general solution, a bandwidth regulation function controls penetration data speed for every flow, and limits regulating the flow of traffic etc., and fabricating is preferred. Drawing 6 is a block diagram showing regulation of a different flow. As for a regulation parameter, it is preferred to be established by defining a KOMITTEDDO information rate (CIR) per byte for every time, and defining both of KOMITTEDDO burst sizes (CBS) and surplus burst sizes (Electronic Broking Systems) per byte. As for a packet, it is preferred to be classified namely, marked on the 1st bucket (drops proper (DE) bucket) 200 and the 2nd bucket (drops bucket) 202.

[0048]When a packet is shown with given entry speed, it is preferred to be marked by the present balance in each bucket and the relation to CBS and Electronic Broking Systems. As for the 1st bucket, it is preferred to maintain abandonment proper (DE) balance. As for the 2nd bucket, it is preferred to maintain drops balance. When entry speed is smaller than CBS, marking on a packet with transmission is preferred. Entry speed is larger than CBS, or although it is equal to it, when smaller than Electronic Broking Systems, marking on a packet with DE is preferred. Entry speed is larger than Electronic Broking Systems, or when equal to it, marking on a packet with drops is preferred.

[0049]Drawing 7 is one embodiment of this invention, and is the regulation data table 250 used in order to regulate a data packet based on two or more policy levels. The regulation data table 250 can be stored in the regulation engine which can suppose that it is the same as that of the regulation engine 166 of drawing 4. The regulation data table 250 can also be called a regulation database.

[0050]The regulation data table 250 contains the regulated data which checks the current speed of the traffic which he follows through packet switching controllers, such as the packet switching controller 130 of drawing 4, for example. Although the regulation data table 250 can be constituted from various methods, it is preferred that

constitute as an entry one by one and each entry provides the regulated data 252 related with the specific policy group. As for each regulated data 252, it is preferred to identify by original regulation identifier (ID) / key 254.

[0051]As for regulation ID254, it is preferred to identify a different policy group who can classify a packet. As for each regulation ID254, it is preferred to comprise the customer identifier 254a and/or the application identifier 254b. As for a customer identifier, it is preferred to identify a specific customer based on a source address, a physical port, etc. As for the application identifier 254b, it is preferred that it is the internal identifier assigned by application RAM based on the type of the application related with the packet. Illustration application contains a Web site application, voice-over IP (VoIP) application, etc.

[0052]It is preferred that the following regulation ID256 enables it to identify the policy group of the addition which telescopic search of a regulation database can apply to Paquette. It is preferred to search the regulated data 252 related with those policy groups, and to carry out present Paquette's speed check.

[0053]As for each regulated data 252, it is preferred that the limit of the present bandwidth and each policy group's bandwidth identified by regulation ID254 is shown. As for the drops balance 252c and the drops proper (DE) balance 252d, it is preferred to maintain the count of the quantity of traffic which progresses through the Paquette switching controller. It is preferred to advise to carry out DE and the mark which transmit present Paquette for the drops balance 252c and the DE balance 252d as compared with the drops limit 252e and 252 f of DE limits, respectively, and to transmit, or to drop immediately. As for the drops balance 252c, not **********ing is preferred until the DE balance 252d becomes larger than 252 f of DE limits.

[0054]As for each regulated data 252, it is preferred that the time stamp 252b in which the time when the last balance calculation was carried out is shown further is included. If the present time and the information on a time stamp are given, it is possible to calculate the traffic speed in this time by measuring the time which has passed since the last balance calculation. The size of the increment of a time stamp can be adjusted based on the value of the budget (CIR) 252a currently too maintained by the regulation data table 250. For example, the budget value can give a definition as a number of bytes per time stamp increment at one embodiment of this invention.

[0055]As for a regulation engine, in the shown regulation data table 250, it is preferred to generate the 1st policy result which shows arrangement of the packet which carried out the speed check 258 or 260 based on 1st regulation ID, and was advised. As for a regulation engine, it is preferred to judge whether a packet is regulated based on an additional policy group. Therefore, as for a policy engine, it is preferred to investigate

following regulation ID field 256 and to search the regulated data identified by ID. Subsequently, it is preferred to carry out the 2nd speed check 262 to the same packet, and to generate the 2nd policy result based on the 2nd speed check. It is possible to continue an additional speed check and to carry out based on the value about following policy ID field 256. In one embodiment of this invention, it is possible to maintain the performance of line speed, carrying out each packet pair and performing a maximum of four regulation algorithms. In other embodiments, it is possible to perform few [that it is more than four or] regulation algorithms.

[0056]Drawing 8 is an illustration flow chart of a multilevel regulatory process. It starts at Step 300 and, as for a process, it is [a regulation engine] preferred to receive new regulation ID to an ingress packet. It is preferred that a regulation engine searches with Step 302 the regulated data related with regulation ID. It is preferred that a regulation engine calculates new drops balance or DE balance by the following desirable formulas at Step 304.

[0057]By a balance $_{new}$= balance $_{old}$-[budget x(time-time stamp)]+ packet size top formula, balance $_{new}$ and balance $_{old}$, It is preferred to express the new balance to the drops bucket or DE bucket related with regulation ID and the present balance, respectively. As for a budget, it is preferred to express the budget 252a related with regulation ID, such as CIR. Present drops balance and DE balance correspond to the drops balance 252c and the DE balance 252d, respectively. As for time and a time stamp, it is preferred to express the time and the time stamp 252b of the present when it is related with regulation ID, respectively. As for a packet size, it is preferred to express the size of the packet currently processed.

[0058]At Step 306, new drops balance or DE balance is applied to the drops limit 252e or 252 f of DE limits. It is preferred to apply this balance to DE balance until it exceeds DE limit. It is preferred to measure DE balance and DE limit, and as for a regulation engine, when DE balance is smaller than DE limit, it is preferred to make a decision which transmits a packet. When DE balance exceeds DE limit, it is preferred to apply this balance to drops balance. Subsequently, it is preferred to measure drops balance and a drops limit, and when drops balance is smaller than a drops limit, it is preferred [a regulation engine] to determine to mark with DE and to transmit a packet. However, as for a regulation engine, when a drops limit is exceeded, it is preferred to determine to discard a packet immediately.

[0059]For example, actually, new balance is calculated and it is preferred to rank second, to measure DE limit and a drops limit, and to determine transmission status. As for balance, being updated based on a transmission result is preferred. For example, when marked on the packet with transmission, it is preferred to update DE balance. That is,

as for DE buckets, such as the 1st bucket 200 of drawing 6, when marked on the packet with transmission, being filled is preferred, for example. As other examples, when marked on the packet with DE, it is preferred to update drops balance. That is, when marked on the packet with DE, drops buckets, such as the 2nd bucket 202 of drawing 6, are filled. DE bucket is already filled at this time. Since both buckets are filled at this time as other examples when marked on the packet with drops, DE balance or drops balance is not updated, either.

[0060]It is judged whether regulation ID of the addition shown to the present packet at Step 308 exists. When it exists, a process returns to Step 302, searches the regulated data identified by additional regulation ID, and generates an additional policy result.

[0061]It is preferred that a regulation engine notifies a policing result to arrangement engines, such as the arrangement engine 144 of drawing 4, at Step 310, for example, and this can be called regulation advice again. In the case where two or more policy results are available to the packet currently processed, as for a regulation engine, it is preferred to choose the most conservative policing result, i.e., the policing result in the case of being the worst, and it is preferred to return the result to an arrangement engine. As for an arrangement engine, it is preferred to use arrangement advice of others, such as a policing result and security advice, and a pattern-matching result, and to generate arrangement determination.

[0062]As for a regulation engine, at Step 312, it is preferred to receive the notice of arrangement determination from an arrangement engine. The arrangement determination can include the determination about whether the packet was transmitted, or it marked with DE and transmitted, or it dropped. It is preferred to judge whether the regulation engine transmitted the packet at Step 314. When that is right, each regulated data related with the transmitted packet is updated at Step 316 reflecting the traffic which increased.

[0063]As for the value updated in a regulation database, it is preferred that DE balance, drops balance, and one or more of a time stamp are included. As for DE balance, being updated when smaller than DE limit is preferred. Drops balance has DE balance larger than DE limit, and it is preferred to be updated when drops balance is smaller than a drops limit. It is preferred that neither is updated, when both balance is over each limit. For example, when packets, such as a frame, are dropped for the arbitrary reasons shown by arrangement determination in all cases, it is desirable to add the value of a "packet size" (size of a packet) to neither of the balance. Thus, it is preferred that an exact count is created about the packet which carries out ingress to an exchange fabric.

[0064]IV. In one embodiment of flow speed regulation this invention provided with deferment debiting, it is preferred to keep unchanged (deferred) and to use debiting

with flow speed regulation. Drawing 9 is the block diagram 400 of the packet switching controller which is provided with deferment debiting and has flow speed regulation in this embodiment of this invention. Deferment debiting can be used together with multilevel regulation logic.

[0065]As shown in drawing 9, the field extracting apparatus 402 receives a packet, keeps flow information unchanged with the general decision logic 408, provides the DEBITTO regulation logic 410 with it, and provides the packet size computing device 404 with a packet. The packet size computing device 404 provides the packet size buffer 406 with an output, and provides the packet buffer 412 with a packet. It keeps unchanged with the general decision logic 408, and the DEBITTO regulation logic 410 provides the arrangement logic 414 with general decision results and a policing result, respectively. Arrangement logic provides the packet buffer 412 with an arranging result. The arrangement logic 414 keeps an arranging result unchanged, and provides the DEBITTO regulation logic 410 with it, and this uses an arranging result and packet size information for deferment debiting.

[0066]Since the customer with the qualification for receiving a different quality of service is vying in the bandwidth of a share network, flow speed regulation is becoming still more important in data-communications networking. Usually, flow speed regulation compares the packet within a flow with one or more bandwidth contracts that the flow was defined, (i) include the thing which discard; (for example, the packet -- the abandonment -- it marks that it is proper) or the (iii) packet which recognizes a packet by;(ii) conditional [which recognizes a packet without conditions] and which is attached [it is alike and] and solved.

[0067]Usually, a flow speed regulation method maintains a "token bucket", and expresses available bandwidth under each bandwidth contract now. Usually, it is considered that a packet is in the bandwidth contract of a flow when sufficient token exists in the bucket currently maintained for the contract now, and when sufficient token does not exist in the bucket currently maintained for the contract now, it is considered that the packet is over a contract. A token is added to a bucket via a time credit as time passes. A token is subtracted from a bucket when a packet is recognized via packet size DEBITTO.

[0068]The general formula used in order to maintain the state of a token bucket is as follows.

[0069]a $TC_{new}=TC_{old}+C-D$ top type -- TC_{new}= -- new token count TC_{old}= -- it is an old token count C= time credit D= SAIZUDE bit.

[0070]It is possible to apply one instance of a token bucket characteristic equation, and to carry out easy recognition/abandonment regulation determination as follows. When

arriving for regulation determination of the packet within a flow, the time credit C reflecting the time which has passed since the regulation determination about a precedence packet is added, By subtracting the SAIZUDE bit D reflecting the size of the present packet, new token count TC_{new} to the bandwidth contract of a flow is calculated. Subsequently, new token count TC_{new} to the bandwidth contract of a flow is compared with zero. When new token count TC_{new} is larger than zero or equal to zero, the present packet is in a bandwidth contract and is recognized. When new token count TC_{new} is smaller than zero, the present packet is over a bandwidth contract and is discarded.

[0071]It is possible to apply two instances of a token bucket characteristic equation to the same flow, and to provide the policing result refined more. For example, it is possible to maintain independently an abandonment token bucket and an abandonment proper token bucket to a flow. In that case, although new abandonment token count $TC_{new\text{-}de}$ is larger than zero or equal to zero, when smaller than zero, the present packet has new abandonment token count $TC_{new\text{-}d}$ in an abandonment bandwidth contract, but it is over an abandonment proper bandwidth contract. therefore, the present packet -- abandonment (since it is over an abandonment proper bandwidth contract) -- it is recognized on condition that it is marked that it is proper (since it is in a drops bandwidth contract). Such a 3 level "double token bucket" regulation method is indicated to IETF Request for Comment 2697 of the name "A Single Rate Three ColorMarker."

[0072]applying a token bucket characteristic equation and regulating high-speed data flow with the packet switching controller of present condition art, subtracted the SAIZUDE bit D which is reflecting the size of the present packet especially, and it was faced with practical difficulty about instruction of making a regulation decision after that. To the 1st, the size of the present packet can be determined in the exterior of regulation logic. Therefore, the SAIZUDE bit D to the present packet may not be available when a regulation decision is made. The last arrangement of a packet may not be directed to the 2nd only by regulation determination. Therefore, the total (deduction) of the SAIZUDE bit D to the present packet may demand to be behind reverse. The 3rd will consider that the present packet exceeds a bandwidth contract, even when token sufficient when the SAIZUDE bit D to the present packet makes a regulation-after total decision to accommodate most (they are not all) packets exists in a bucket.

[0073]Since data transfer speed is more nearly exponentially [than the maximum packet size] large by a high-speed controller, the practical advantage which deducts the SAIZUDE bit D to the present packet, and makes a regulation decision after that on the other hand is not clear. As long as the SAIZUDE bit D is created by within a time [behind moderate], it is a grade to which it is on the title to a flow at most, and a

temporary bandwidth breach of contract is carried out.

[0074]In this embodiment of this invention, it is preferred to conquer the above-mentioned difficulty about using deferment debiting and regulating high-speed data flow with the application of a general token bucket characteristic equation.

[0075]For example, it is possible to provide the data regulation method. The data regulation method, ; which receives a packet -- a time credit being added to the 1st token count, and. ; which generates the 2nd token count -- the policing result to; packet which generates the policing result to a packet with the application of the 2nd token count is applied, a SAIZUDE bit is subtracted from the 2nd token count, and the 3rd token count is generated. Or it is preferred that what a SAIZUDE bit is not subtracted from the 2nd token count, and the 3rd token count is not generated for is included.

[0076]The data regulation method can have what the policing result to the 2nd packet is generated for with the application of the; 4th token count which adds; time credit which receives the 2nd packet to the 2nd token count, and generates the 4th token count.

[0077]It is possible to provide other data regulation methods. This data regulation method, ; which receives a packet -- a time credit being added to the 1st token count, and. ; which generates the 2nd token count -- with the application of the 2nd token count. ; which generates the policing result to a packet -- the arranging result to; packet which generates the arranging result to a packet with the application of the policing result to a packet is applied, a SAIZUDE bit is subtracted from the 2nd token count, and the 3rd token count is generated. Or it is preferred that what a SAIZUDE bit is not subtracted from the 2nd token count, and the 3rd token count is not generated for is included.

[0078]It is possible to apply a policing result in this data regulation method as advice which has other at least one advice, and to generate the arranging result to a packet.

[0079]; in which other data regulation methods receive a packet -- a time credit being added to each of a token count, and. ; which generates each of the 2nd token count -- with the application of each of the 2nd token count. ; which generates the policing result to a packet -- the policing result to a packet is applied, a SAIZUDE bit is subtracted from at least one of the 2nd token counts, and at least one 3rd token count is generated. Or it is preferred that what a SAIZUDE bit is not subtracted from at least one of the 2nd token counts, and at least one 3rd token count is not generated for is included.

[0080]Other data regulation methods, ; which receives a packet -- a time credit being added to each of a token count, and. ; which generates each of the 2nd token count -- with the application of each of the 2nd token count. ; which generates the policing result to a packet -- with the application of the policing result to a packet. ; which generates the arranging result to a packet -- the arranging result to a packet is applied, a

SAIZUDE bit is subtracted from at least one of the 2nd token counts, and at least one 3rd token count is generated. Or it is preferred that what a SAIZUDE bit is not subtracted from at least one of the 2nd token counts, and at least one 3rd token count is not generated for is included.

[0081]The following data regulation methods show the flow speed regulation provided with deferment debiting in one embodiment of this invention further.

[0082]The data regulation method adds; time credit which receives a packet to the 1st token count, ; which generates the 2nd token count --; policing result which generates the policing result to a packet with the application of the 2nd token count, [apply and] It is preferred that what subtract a SAIZUDE bit from the 2nd token count, and the 3rd token count is generated, or a SAIZUDE bit is not subtracted from the 2nd token count, and the 3rd token count is not generated for is included.

[0083]As for this data regulation method, it is preferred that what the policing result to the 2nd packet is generated for with the application of the; 4th token count which adds further; time credit which receives the 2nd packet to the 2nd token count, and generates the 4th token count is included.

[0084]Other data regulation methods, ; which receives a packet -- a time credit being added to the 1st token count, and. ; which generates the 2nd token count -- with the application of the 2nd token count. ; which generates the policing result to a packet --; arranging result which generates the arranging result to a packet with the application of a policing result is applied, a SAIZUDE bit is subtracted from the 2nd token count, and the 3rd token count is generated. Or it is preferred that what a SAIZUDE bit is not subtracted from the 2nd token count, and the 3rd token count is not generated for is included. It is possible to generate an arranging result with the application of a policing result as advice which has other at least one advice.

[0085]Other data regulation methods, ; which receives a packet -- a time credit being added to each of a token count, and. ; which generates each of the 2nd token count -- with the application of each of the 2nd token count. ; which generates the policing result to a packet -- a policing result is applied, a SAIZUDE bit is subtracted from at least one of the 2nd token counts, and at least one 3rd token count is generated. Or it is preferred that what a SAIZUDE bit is not subtracted from at least one of the 2nd token counts, and at least one 3rd token count is not generated for is included.

[0086]Other data regulation methods, ; which receives a packet -- a time credit being added to each of a token count, and. ; which generates each of the 2nd token count -- with the application of each of the 2nd token count. ; which generates the policing result to a packet --; arranging result which generates the arranging result to a packet with the application of a policing result is applied, a SAIZUDE bit is subtracted from at least

one of the 2nd token counts, and at least one 3rd token count is generated. Or it is preferred that what a SAIZUDE bit is not subtracted from at least one of the 2nd token counts, and at least one 3rd token count is not generated for is included.

[0087]Probably, there is no difficulty in devising a modification gestalt in any way, without never deviating from the range and pneuma of this invention, if it is a person skilled in the art, although this invention was explained about a certain specific embodiment. Therefore, this invention should understand that it is possible to perform by the method except having explained in detail. Therefore, it should be considered that this embodiment of this invention is illustration-like at all points, and is not restrictive, and the range of this invention is shown by not the above-mentioned explanation but an attached claim and equivalent.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1]It is a figure of the network environment containing a packet switching node which uses one embodiment of this invention.

[Drawing 2]It is a block diagram of the exchange interface in one embodiment of this invention.

[Drawing 3]It is a block diagram of the programmable Paquette switching controller in one embodiment of this invention.

[Drawing 4]It is a block diagram of the packet switching controller which has programmable arrangement logic in one embodiment of this invention.

[Drawing 5]It is a process-flow figure which can be set to one embodiment of this invention and which generates arrangement determination by a program using two or more arrangement advice and classification information.

[Drawing 6]It is a block diagram showing the process of marking a packet on a different classification.

[Drawing 7]It is a regulation data table used in order to regulate a data packet based on two or more policy levels which can be set to one embodiment of this invention.

[Drawing 8]It is a flow chart of a multilevel regulatory process in one embodiment of this invention.

[Drawing 9]It is a block diagram of the packet switching controller which has the flow speed regulation provided with deferment debiting in one embodiment of this invention.

[Description of Notations]

100 Programmable packet switching controller

102, 132, and 412 Packet buffer
104, 134 packet classification engine
106 and 138 Application engine
120 and 166 Regulation engine
122 and 170 Regulation advice
124 Regulation ID
130 Packet switching controller
136 Pattern-matching search logic
140 Sauce searching engine
142 Address searching engine
144 Arrangement engine
146 Inbound packet
148 and 150 An inbound packet or its part
152 Program discernment
154 Pattern-matching result
156 Output
160 Arrangement advice
162 Arrangement determination
164 Outbound packet
168 Regulation identifier
200 The 1st bucket (drops proper (DE) bucket)
202 The 2nd bucket (drops bucket)
250 Regulation data table
252 Regulated data
252a budget (CIR)
252b Time stamp
252c Drops balance
252 d Drops proper (DE) balance
252e Drops limit
252f DE limit
254 Regulation identifier (ID) / key
254a Customer identifier
254b Application identifier
256 The following regulation ID field
258 and 260 Speed check
262 The 2nd speed check
402 Field extracting apparatus

404 Packet size computing device

406 Packet size buffer

408 General decision logic

410 Deferment DEBITTO regulation logic

414 Arrangement logic

---

(19)日本国特許庁（ＪＰ）　(12)**公開特許公報（Ａ）**　(11)特許出願公開番号
特開2002－44150
（P2002－44150A）
(43)公開日　平成14年２月８日(2002.2.8)

(51)Int.Cl.[7]	識別記号	ＦＩ	ﾃｰﾏｺｰﾄﾞ(参考)
Ｈ０４Ｌ　12/56	２００	Ｈ０４Ｌ　12/56	２００Ｂ　５Ｋ０３０

審査請求　未請求　請求項の数46　ＯＬ　外国語出願　（全 67 頁）

(21)出願番号　特願2001－154076(P2001－154076)
(22)出願日　平成13年５月23日(2001.5.23)
(31)優先権主張番号　２０６６１７
(32)優先日　平成12年５月24日(2000.5.24)
(33)優先権主張国　米国（ＵＳ）
(31)優先権主張番号　２０６９９６
(32)優先日　平成12年５月24日(2000.5.24)
(33)優先権主張国　米国（ＵＳ）
(31)優先権主張番号　２２０３３５
(32)優先日　平成12年７月24日(2000.7.24)
(33)優先権主張国　米国（ＵＳ）

(71)出願人　501205005
アルカテル・インターネツトワーキング（ピー・イー）インコーポレイテツド
アメリカ合衆国、ワシントン・99206、スポーケン、イースト・スプレーグ・アベニユー・11707
(72)発明者　マシユー・トーレガス
アメリカ合衆国、ワシントン・99037、ベラデール、イースト・トウエンテイ・サード・アベニユー・16521
(74)代理人　100062007
弁理士　川口　義雄　（外１名）

最終頁に続く

(54)【発明の名称】　マルチレベル規制論理を有するパケットプロセッサ

(57)【要約】
【課題】　バックプレーンと複数のパケットプロセッサを含むスイッチを提供すること。
【解決手段】　１つまたは複数のパケットプロセッサは、マルチレベル規制論理を含む。パケットプロセッサは、パケットを受信して、パケットを複数の規制可能グループに分類する。パケットを、規制可能グループについて定義された帯域幅契約と比較する。入れ子式探索を規制データベースでパケットに対して実施して、複数のグループを識別し、複数の規制可能グループに対する規制データを検索する。この規制結果は、最悪の場合の規制結果を取り入れることによって、１つの規制結果に組み合わせることが可能であり、これを勧告として配置論理に適用し、他の配置勧告と組み合わせて、パケットに対する配置決定を行う。

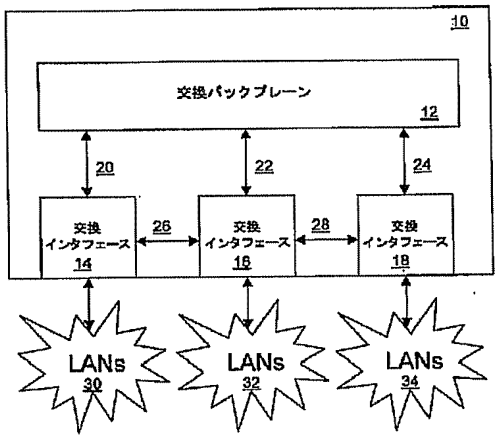

FIG. 1

1

【特許請求の範囲】

【請求項１】　パケットを受信する入力部と、
パケットを複数の規制可能グループに分類する規制エレメントとを備え、
パケットが、規制可能グループについて定義された１つまたは複数の帯域幅契約と比較され、１つまたは複数の規制結果を生成する、パケット交換コントローラ。

【請求項２】　規制エレメントが規制データベースを含み、第１規制データと第２規制可能グループ識別子を検索するために第１規制可能グループ識別子が規制データベースに適用され、第１規制結果を生成するために第１規制データが適用され、第２規制データを検索するために第２規制可能グループ識別子が規制データベースに適用され、第２規制結果を生成するために第２規制データが適用される、請求項１に記載のパケット交換コントローラ。

【請求項３】　パケットに対する配置決定を行う配置エンジンをさらに備え、規制結果が、１つまたは複数の配置勧告を含み、配置エンジンが、規制結果と少なくとも１つの他の配置勧告を使用して、パケットに対する配置決定を行う、請求項１に記載のパケット交換コントローラ。

【請求項４】　最悪の場合の規制結果を取り入れることによって、規制結果を１つの結果に組み合わせる、請求項１に記載のパケット交換コントローラ。

【請求項５】　規制エレメントを使用してパケットを処理する方法であって、
パケットを受信するステップと、
パケットを複数の規制可能グループに分類するステップと、
パケットを、規制可能グループについて定義された１つまたは複数の帯域幅契約と比較して、１つまたは複数の規制結果を生成するステップとを含む、パケットを処理する方法。

【請求項６】　規制エレメントが、規制データベースを含み、
第１規制可能グループ識別子を規制データベースに適用して、第１規制データと第２規制可能グループ識別子を検索するステップと、
第１規制データを使用して、第１規制結果を生成するステップと、
第２規制可能グループ識別子を規制データベースに適用して、第２規制データを検索するステップと、
第２規制データを使用して、第２規制結果を生成するステップとをさらに含む、請求項５に記載のパケットを処理する方法。

【請求項７】　規制結果が、１つまたは複数の配置勧告を含み、規制結果と少なくとも１つの他の配置勧告を使用して、パケットに対する配置決定を行うステップをさらに備える、請求項５に記載のパケットを処理する方法。

2

【請求項８】　最悪の場合の規制結果を取り入れることによって、規制結果を１つの結果に組み合わせるステップをさらに含む、請求項５に記載のパケットを処理する方法。

【請求項９】　データ通信スイッチが受信したデータパケットを規制する方法であって、
データパケットを複数の規制可能グループに分類するステップと、
１つまたは複数の規制可能グループに関連付けられている規制データを識別するステップと、
規制データを適用して、規制可能グループに対する１つまたは複数の規制結果を生成するステップと、
規制結果から、データパケットの配置を勧告するステップとを含む方法。

【請求項１０】　特定の規制可能グループが、規制するアプリケーションのタイプを識別する、請求項９に記載の方法。

【請求項１１】　規制データが、少なくとも１つの規制可能グループについて指定された帯域幅制約に関する情報を含む、請求項９に記載の方法。

【請求項１２】　規制結果が、データパケットを転送すべきかどうかを示す、請求項９に記載の方法。

【請求項１３】　規制結果が、データパケットがドロップするのに適格であるかを示す、請求項９に記載の方法。

【請求項１４】　規制結果が、データパケットをドロップすべきかどうかを示す、請求項９に記載の方法。

【請求項１５】　配置を勧告するステップが、規制結果を組み合わせて、勧告を作成するステップを含む、請求項９に記載の方法。

【請求項１６】　配置を勧告するステップが、勧告された配置として規制結果の１つを選択することを含む、請求項９に記載の方法。

【請求項１７】　勧告された配置に基づいて、規制データを更新するステップをさらに含む、請求項９に記載の方法。

【請求項１８】　データ通信スイッチが受信したデータパケットを規制する方法であって、
複数の規制可能グループに対する規制データを指定する複数の規制データエントリを含む規制データベースを創出するステップと、
第１識別子を適用して、第１規制可能グループに関連付けられている第１規制データと、第２規制可能グループを識別する第２識別子を検索するステップと、
第１規制データを適用して、第１規制結果を生成するステップと、
第２識別子を適用して、第２規制データを検索するステップと、
第２規制データを適用して、第２規制結果を生成するス

テップと、
第１および第２規制結果から、データパケットの配置を勧告するステップとを含む方法。
【請求項１９】　特定の規制可能グループが、規制するアプリケーションのタイプを識別する、請求項１８に記載の方法。
【請求項２０】　規制データが、規制可能グループについて指定された帯域幅制約に関する情報を含む、請求項１８に記載の方法。
【請求項２１】　規制結果が、データパケットを転送すべきかどうかを示す、請求項１８に記載の方法。
【請求項２２】　規制結果が、データパケットがドロップするのに適格であるかを示す、請求項１８に記載の方法。
【請求項２３】　規制結果が、データパケットをドロップすべきかどうかを示す、請求項１８に記載の方法。
【請求項２４】　配置を勧告するステップが、第１および第２規制結果を組み合わせて勧告を作成する、請求項１８に記載の方法。
【請求項２５】　配置を勧告するステップがさらに、勧告された配置として、第１または第２規制結果のどちらかを選択することを含む、請求項１８に記載の方法。
【請求項２６】　勧告された配置に基づいて、第１または第２規制データを更新するステップをさらに含む、請求項１８に記載の方法。
【請求項２７】　規制エンジンが、パケットを複数の規制可能グループに分類し、規制可能グループのそれぞれについてパケットを、帯域幅契約のそれぞれと比較して、規制結果のそれぞれを生成する、データ通信ノードのための規制エンジン。
【請求項２８】　第１規制データと第２規制可能グループ識別子を検索するために第１規制可能グループ識別子が規制データベースに適用され、第１規制結果を生成するために第１規制データが適用され、第２規制データを検索するために第２規制可能グループ識別子が規制データベースに適用され、第２規制結果を生成するために第２規制データが適用される、データ通信ノードのための規制エンジン。
【請求項２９】　パケットを受信する入力部と、
パケットを複数の規制可能グループに分類する規制手段とを備え、
パケットが、規制可能グループについて定義された１つまたは複数の帯域幅契約と比較され、１つまたは複数の規制結果を生成するパケットプロセッサ。
【請求項３０】　規制手段が、規制データベースを含み、第１規制データと第２規制可能グループ識別子を検索するために第１規制可能グループ識別子が規制データベースに適用され、第１規制結果を生成するために第１規制データが適用され、第２規制データを検索するために第２規制可能グループ識別子が規制データベースに適用され、第２規制結果を生成するために第２規制データが適用される、請求項２９に記載のパケットプロセッサ。
【請求項３１】　パケットに対する配置決定を行う配置手段をさらに備え、規制結果が１つまたは複数の配置勧告を含み、配置手段が、規制結果と少なくとも１つの他の配置勧告を使用してパケットに対する配置決定を行う、請求項２９に記載のパケットプロセッサ。
【請求項３２】　最悪の場合の規制結果を取り入れることによって、規制結果を１つの結果に組み合わせる、請求項２９に記載のパケットプロセッサ。
【請求項３３】　パケット交換コントローラが、デビッティングエレメントをさらに備え、少なくとも１つの帯域幅契約が、前記帯域幅契約の下で利用可能な帯域幅を示す関連トークンバケットを有し、デビッティングエレメントが、規制結果を使用して、関連トークンバケットにデビットするかどうかを判定する、請求項１に記載のパケット交換コントローラ。
【請求項３４】　パケット交換コントローラが、デビッティングエレメントをさらに備え、少なくとも１つの帯域幅契約が、帯域幅契約の下で利用可能な帯域幅を示す関連トークンバケットを有し、配置エンジンが関連トークンバケットにデビットするか否かを判定するために使用するデビッティングエレメントに配置決定を提供するまで、デビッティングエレメントが、バケットサイズを関連トークンバケットにデビットすることを据え置く、請求項３に記載のパケット交換コントローラ。
【請求項３５】　少なくとも１つの帯域幅契約が、帯域幅契約の下で利用可能な帯域幅を示す関連トークンバケットを有し、規制結果を使用して関連トークンバケットにデビットするか否かを判定することをさらに含む、請求項５に記載のパケットを処理する方法。
【請求項３６】　少なくとも１つの帯域幅契約が、帯域幅契約の下で利用可能な帯域幅を示す関連トークンバケットを有し、配置決定を使用してパケットサイズを関連トークンバケットにデビットするか否かを判定することをさらに含む、請求項７に記載のパケットを処理する方法。
【請求項３７】　規制結果からの配置勧告と少なくとも１つの他の配置勧告を使用して、データパケットに対する配置決定を生成するステップと、
配置決定を使用して帯域幅制約に関する情報を更新するか否かを判定するステップとをさらに含む、請求項１１に記載のデータパケットを規制する方法。
【請求項３８】　第１および第２規制結果からの配置勧告と少なくとも１つの他の配置勧告を使用して、データパケットに対する配置決定を生成するステップと、
配置決定を使用して帯域幅制約に関する情報を更新するか否かを判定するステップとをさらに含む、請求項２０に記載のデータパケットを規制する方法。

【請求項３９】　帯域幅契約の下で利用可能な帯域幅を更新するか否かを、規制結果に基づいて判定する、請求項２７に記載の規制エンジン。

【請求項４０】　パケットプロセッサがさらにデビット手段を備え、少なくとも１つの帯域幅契約が、帯域幅契約の下で利用可能な帯域幅を示す関連トークンバケットを有し、配置手段が関連トークンバケットにデビットするか否かを判定するために使用するデビッティング手段に配置決定を提供するまで、デビッティング手段が、バケットサイズを関連トークンバケットにデビットすることを据え置く、請求項３１に記載のパケットプロセッサ。

【請求項４１】　パケットを受信するステップと、
時間クレジットを第１トークンカウントに追加して、第２トークンカウントを生成するステップと、
第２トークンカウントを適用して、パケットに対する規制結果を生成するステップと、
規制結果を適用し、第２トークンカウントからサイズデビットを減算して第３トークンカウントを生成するか否かを判定するステップと、
減算が規制結果を適用することにより判定されている場合に第２トークンカウントからサイズデビットを減算して第３トークンカウントを生成するステップとを備える、データ規制方法。

【請求項４２】　第２パケットを受信するステップと、
第３トークンカウントが生成されていない場合に第２の時間クレジットを第２トークンカウントに追加して第４トークンカウントを生成するステップと、
第３トークンカウントが生成されている場合に第２の時間クレジットを第３トークンカウントに追加して第４トークンカウントを生成するステップと、
第４トークンカウントを適用して、第２パケットに対する規制結果を生成するステップとをさらに含む、請求項４１に記載のデータ規制方法。

【請求項４３】　パケットを受信するステップと、
時間クレジットを第１トークンカウントに追加して第２トークンカウントを生成するステップと、
第２トークンカウントを適用してパケットに対する規制結果を生成するステップと、
規制結果を適用してパケットに対する配置結果を生成するステップと、
配置結果を適用し、第２トークンカウントからサイズデビットを減算して第３トークンカウントを生成するか否かを判定するステップと、
減算が配置結果を適用することにより判定されている場合に第２トークンカウントからサイズデビットを減算して第３トークンカウントを生成するステップとを含む、データ規制方法。

【請求項４４】　少なくとも１つの他の勧告を有する勧告として規制結果を適用して、配置結果を生成する、請求項４３に記載のデータ規制方法。

【請求項４５】　パケットを受信するステップと、
時間クレジットをトークンカウントのそれぞれに追加して第２トークンカウントのそれぞれを生成するステップと、
第２トークンカウントのそれぞれを適用してパケットに対する規制結果を生成するステップと、
規制結果を適用し、第２トークンカウントの少なくとも１つからサイズデビットを減算して少なくとも１つの第３トークンカウントを生成するか否かを判定するステップと、
減算が規制結果を適用することにより判定されている場合、第２トークンカウントの少なくとも１つからサイズデビットを減算して少なくとも１つの第３トークンカウントを生成するステップとを含む、データ規制方法。

【請求項４６】　パケットを受信するステップと、
時間クレジットをトークンカウントのそれぞれに追加して第２トークンカウントのそれぞれを生成するステップと、
第２トークンカウントのそれぞれを適用してパケットに対する規制結果を生成するステップと、
規制結果を適用してパケットに対する配置結果を生成するステップと、
配置結果を適用して第２トークンカウントの少なくとも１つからサイズデビットを減算して、少なくとも１つの第３トークンカウントを生成するか否かを判定するステップと、
減算が配置結果を適用することにより判定されている場合、第２トークンカウントの少なくとも１つからサイズデビットを減算して少なくとも１つの第３トークンカウントを生成するステップとを含む、データ規制方法。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】関連出願の相互参照

本出願は、２０００年５月２４日に出願された、「Ｓｙｓｔｅｍ　ａｎｄ　Ｍｅｔｈｏｄ　ｆｏｒ　Ｅｎｈａｎｃｅｄ　Ｌｉｎｅ　Ｃａｒｄｓ」という名称の米国仮出願第６０／２０６，６１７号、２０００年５月２４日に出願された「Ｆｌｏｗ　Ｒｅｓｏｌｕｔｉｏｎ　Ｌｏｇｉｃ　Ｓｙｓｔｅｍ　ａｎｄ　Ｍｅｔｈｏｄ」という名称の米国仮出願第６０／２０６，９９６号、および２０００年７月２４日に出願された「Ｐｒｏｇｒａｍｍａｂｌｅ　Ｐａｃｋｅｔ　Ｐｒｏｃｅｓｓｏｒ」という名称の米国仮出願第６０／２２０，３３５号の優先権を請求するものであり、これら全ての内容は、参照によって、本明細書に完全に組み込まれている。本出願は、２０００年１２月２８日に出願された「Ｐｒｏｇｒａｍｍａｂｌｅ　Ｐａｃｋｅｔ　Ｐｒｏｃｅｓｓｏｒ　ｗｉｔｈ　Ｆｌｏｗ　Ｒｅｓｏｌｕｔｉｏｎ　Ｌｏｇｉｃ」という名称の米国特許出願第０９／７５１，１９４号に開示さ

れている主題に関係する主題を含む。この内容は、参照により完全に本明細書に組み込まれている。

【０００２】本発明は、一般にデータ通信スイッチに関し、より詳細には、データパケットに対する複数レベルの速度規制（ｒａｔｅ　ｐｏｌｉｃｉｎｇ）を使用するデータ通信スイッチに関する。

【０００３】

【従来の技術】異なるサービス品質（ＱｏＳ）を受ける資格がある顧客が、共通セットであるネットワーク資源の利用可能な帯域幅を競い合っているので、速度規制は、データ通信ネットワークにおいてますます重要になってきている。通常、速度規制は、各パケットを１つのポリシーグループに分類し、分類したパケットを、グループについて定義された１つまたは複数の帯域幅契約と比較することによって、各スイッチにおいて達成される。識別した帯域幅契約に基づいて、パケットを転送すること、廃棄適格（ｄｉｓｃａｒｄ　ｅｌｉｇｉｂｌｅ）のマークをつけて転送すること、または廃棄することが可能である。

【０００４】

【発明が解決しようとする課題】既存の速度規制方法は、通常、トラフィックに関する他の情報に関係なく、ポート毎にデータのトラフィックを規制する。通常、顧客が申し込んだ速度を超えるデータは、輻輳が生じる場合にはドロップされるべきとマークされる。したがって、顧客は、通常、データに関連する特定のアプリケーションに基づいてなど、データのタイプに基づいて、ある種のデータを選択的にドロップするという柔軟性を有していない。

【０００５】通信ネットワークを顧客の個別化した要求に合わせるという要望が強くなってきているので、柔軟性は増大しているが、回線速度を著しく低減するほど実施が複雑でない規制論理を提供することが望ましい。

【０００６】

【課題を解決するための手段】本発明の一実施形態では、パケット交換コントローラが提供される。パケット交換コントローラは、パケットを受信する入力部と、パケットを複数の規制可能グループに分類する規制エレメントを含む。パケットは、規制可能グループについて定義された１つまたは複数の帯域幅契約と比較され、１つまたは複数の規制結果を生成する。

【０００７】本発明の他の実施形態では、パケットを処理する方法が提供される。パケットを受信して、複数の規制可能グループに分類する。パケットを、規制可能グループについて定義された１つまたは複数の帯域幅契約と比較して、１つまたは複数の規制結果を生成する。

【０００８】本発明の他の実施形態では、データ通信スイッチが受信したデータパケットを規制する方法が提供される。データパケットを、複数の規制可能グループに分類する。次いで、１つまたは複数の規制可能グループに関連付けられている規制データを識別する。規制データを適用して、規制可能グループに対する１つまたは複数の規制結果を生成し、規制結果から、データパケットの配置（ｄｉｓｐｏｓｉｔｉｏｎ）を勧告する。

【０００９】本発明の他の実施形態では、データ通信スイッチが受信したデータパケットを規制する方法が提供される。複数の規制可能グループに対して規制データを指定する複数の規制データエントリを含んでいる規制データベースがつくられる。第１識別子を適用して、第１規制可能グループに関連付けられている第１規制データと、第２規制可能グループを識別する第２識別子を検索する。次いで、第１規制データを適用して、第１規制結果を生成する。さらに、第２識別子を適用して、第２規制データを検索する。次いで、第２規制データを適用して、第２規制結果を生成する。第１および第２規制結果から、データパケットの配置を勧告する。

【００１０】本発明の他の実施形態では、データ通信ノードのための規制エンジンが提供される。規制エンジンは、パケットを複数の規制可能グループに分類する。規制可能グループのそれぞれについてパケットを、帯域幅契約のそれぞれと比較して、それぞれの規制結果を生成する。

【００１１】本発明の他の実施形態では、データ通信ノードのための規制エンジンが提供される。第１規制データと第２規制可能グループ識別子を検索するために第１規制可能グループ識別子が規制データベースに適用される。第１規制結果を生成するために第１規制データが適用され、第２規制データを検索するために第２規制可能グループ識別子が規制データベースに適用される。第２規制結果を生成するために第２規制データが適用される。

【００１２】本発明の他の実施形態では、パケットプロセッサが提供される。パケットプロセッサは、パケットを検索する入力部と、パケットを複数の規制可能グループに分類する規制手段を含む。パケットは、規制可能グループについて定義された１つまたは複数の帯域幅契約と比較され、１つまたは複数の規制結果を生成する。

【００１３】

【発明の実施の形態】Ｉ．概略

図１では、パケット交換ノード１０を含むネットワーク環境が示されている。また、パケット交換ノードは、スイッチ、データ通信ノード、またはデータ通信スイッチと呼ぶことが可能である。パケット交換ノード１０は、それぞれＬＡＮ３０、３２、３４に相互接続され、交換バックプレーン１２を介して、データ経路２０、２２、２４により互いに相互接続されている、交換インタフェース１４、１６、および１８を含む。交換バックプレーン１２は、交換ファブリックを含むことが好ましい。また、交換インタフェースは、制御経路２６と２８により、互いに結合することが可能である。

【００１４】交換インタフェース１４、１６、１８は、媒体アクセス制御（ＭＡＣ）ブリッジングおよびインターネットプロトコル（ＩＰ）ルーティングなど、１つまたは複数の操作可能通信プロトコルに従って、ＬＡＮ３０、３２、３４のそれぞれのグループにパケットを送り、またそこからパケットを送ることが好ましい。交換ノード１０は、単に例示の目的で示している。実際には、パケット交換ノードは、３つを超えるまたは３つ未満の交換インタフェースを含むことが可能である。

【００１５】図２は、本発明の一実施形態における交換インタフェース５０のブロック図である。交換インタフェース５０は、図１の交換インタフェース１４、１６、１８などと同様とすることが可能である。交換インタフェース５０は、ＬＡＮとパケット交換コントローラ５２の間に結合されているアクセスコントローラ５４を含む。アクセスコントローラ５４は、例えば媒体アクセスコントローラ（ＭＡＣ）を含むことが可能であり、ＬＡＮを離れたインバウンドパケットを受信し、インバウンドパケットに対してフローに依存しない物理層およびＭＡＣ層操作を実施して、インバウンドパケットをパケット交換コントローラ５２に送信し、フローに依存する処理をすることが好ましい。また、アクセスコントローラ５４は、パケット交換コントローラ５２からアウトバウンドパケットを受信し、パケットをＬＡＮ上で送信することが好ましい。また、アクセスコントローラ５４は、アウトバウンドパケットに対して物理的操作およびＭＡＣ層操作を実施して、その後ＬＡＮ上で送信することが可能である。

【００１６】パケット交換コントローラ５２は、広く多様な通信プロトコルを有するパケットに対処するために、プログラム可能であることが好ましい。パケット交換コントローラ５２は、インバウンドパケットを受信し、パケットを分類し、フロー情報に従ってパケットを修正し、変更されたパケットを、図１の交換バックプレーン１２など交換バックプレーン上で送信することが好ましい。また、パケット交換コントローラ５２は、他のパケット交換コントローラによって修正されたパケットを交換バックプレーンを介して受信し、それをアクセスコントローラ５４に送信してＬＡＮ上で進めることが好ましい。また、パケット交換コントローラ５２は、パケットの選択したものに出口処理（ｅｇｒｅｓｓ　ｐｒｏｃｅｓｓｉｎｇ）を施し、その後アクセスコントローラ５４に送信してＬＡＮ上で転送することが可能である。

【００１７】図３は、本発明の一実施形態におけるプログラム可能パケット交換コントローラ１００のブロック図である。プログラム可能パケット交換コントローラ１００は、例えば、図２のパケット交換コントローラ５２と同様とすることが可能である。プログラム可能パケット交換コントローラ１００は、パケットの入来フローを分類および経路指定するフロー解析論理を有することが好ましい。プログラム可能な性質のために、プログラム可能パケット交換コントローラは、多くの異なるプロトコルおよび／またはフィールドの更新可能性に対処する柔軟性を提供することが好ましい。また、プログラム可能パケット交換コントローラは、パケット交換コントローラ、交換コントローラ、プログラムパケットプロセッサ、ネットワークプロセッサ、通信プロセッサ、または当業者によって一般的に使用されている他の名称で呼ぶことが可能である。

【００１８】プログラム可能パケット交換コントローラ１００は、パケットバッファ１０２、パケット分類エンジン１０４、アプリケーションエンジン１０６、および規制エンジン１２０を含む。規制エンジンは、規制エレメントと呼ぶことも可能である。他の実施形態のパケット交換コントローラは、より多いまたは少ない構成エレメントを含むことが可能である。例えば、他の実施形態のパケット交換コントローラは、パケットの一部を所定のパターンと比較して整合性を調べる、パターン整合モジュールを含むことが可能である。他の実施形態のパケット交換コントローラは、インバウンドパケットを編集して、アウトバウンドパケットを生成する編集モジュールを含むことが可能である。

【００１９】プログラム可能パケット交換コントローラ１００は、インバウンドパケット１０８を受信することが好ましい。パケットは、イーサネット（登録商標）フレーム、ＡＴＭセル、ＴＣＰ／ＩＰおよび／またはＵＤＰ／ＩＰパケットを含むことが可能であるが、これに限定されるものでない。また、他の層２（データリンク／ＭＡＣ層）、層３（ネットワーク層）、または層４（トランスポート層）のデータユニットを含むことが可能である。例えば、パケットバッファ１０２は、イーサネットを介して、１つまたは複数の媒体アクセス制御（ＭＡＣ）層インタフェースからインバウンドパケットを受信することが可能である。

【００２０】受信したパケットは、パケットバッファ１０２に格納されることが好ましい。パケットバッファ１０２は、パケットを受信し、一時的に格納するパケットＦＩＦＯを含むことが可能である。パケットバッファ１０２は、格納されたパケットまたはその一部を、パケット分類エンジン１０４とアプリケーションエンジン１０６に提供して処理することが好ましい。

【００２１】また、パケットバッファ１０２は、パケットを編集して、その後アウトバウンドパケット１１８として交換コントローラから外に進める編集モジュールを含むことが可能である。編集モジュールは、実時間で編集プログラムを作成する編集プログラムビルドエンジン、および／またはパケットを修正する編集エンジンを含むことが可能である。アプリケーションエンジン１０６は、パケットの配置決定を含むことが可能である、アプリケーションデータ１１６を、パケットバッファ１０

２に提供することが好ましい。編集プログラムビルドエンジンは、アプリケーションデータを使用して、編集プログラムを創出することが好ましい。アウトバウンドパケット１１８は、交換ファブリックインタフェースを介して、イーサネットなど、通信ネットワークに送信することが可能である。

【００２２】また、パケットバッファ１０２は、ヘッダデータ抽出器とヘッダデータキャッシュのどちらかまたは両方を含むことが可能である。ヘッダデータ抽出器を使用して、パケットから１つまたは複数のフィールドを抽出し、抽出したフィールドを、抽出ヘッダデータとしてヘッダデータキャッシュに格納することが好ましい。抽出ヘッダデータは、パケットヘッダの一部または全てを含むことが可能であるが、これに限定されるものではない。例えば、イーサネットシステムでは、ヘッダデータキャッシュが、各フレームの初めのＮバイトを格納することも可能である。

【００２３】抽出ヘッダデータは、出力信号１１０としてパケット分類エンジン１０４に提供し、処理することが好ましい。また、アプリケーションエンジンは、インタフェース１１４を介して、抽出ヘッダデータを要求および受信することが可能である。抽出ヘッダデータは、層２のＭＡＣアドレス、８０２．１Ｐ／Ｑタグステータス、層２の密閉（ｅｎｃａｐｓｕｌａｔｉｏｎ）タイプ、層３のプロトコルタイプ、層３のアドレス、ＴｏＳ（サービスのタイプ）値、および層４のポート番号の１つまたは複数を含むことが可能であるが、これに限定されるものではない。他の実施形態では、出力信号１１０は、抽出したヘッダデータの代わりに、またはそれの他に、インバウンドパケット全体を含むことが可能である。他の実施形態では、パケット分類エンジン１０４を使用して、抽出ヘッダデータを、アプリケーションエンジンの使用に適したフォーマットにし、および／またはデータをヘッダデータキャッシュにロードすることが可能である。

【００２４】パケット分類エンジン１０４は、プログラム可能マイクロコード駆動埋込み型処理エンジンを含むことが好ましい。パケット分類エンジン１０４は、命令ＲＡＭ（ＩＲＡＭ）（図示せず）に結合されていることが好ましい。パケット分類エンジンは、ＩＲＡＭに格納されている命令を読み取り、実行することが好ましい。一実施形態では、パケット分類エンジンが実行する命令の多くは、条件付きジャンプである。この実施形態では、分類論理は、異なるタイプのパケット分類を示すことが好ましい葉を端部分に有する決定ツリーを含む。さらに、決定ツリーのブランチは、命令条件と、ヘッダデータキャッシュに格納されているヘッダフィールドとの比較に基づいて選択されることが好ましい。他の実施形態では、分類論理は、決定ツリーに基づいていないことが可能である。

【００２５】本発明の一実施形態では、アプリケーションエンジン１０６は、複数のプログラム可能サブエンジンが直線的にパイプライン化されている、パイプラインアーキテクチャを有することが好ましい。各プログラム可能サブエンジンは、パケットに対する操作を実施することが好ましく、「バケツリレー」方式で、パケットを次のプログラム可能サブエンジンに転送することが好ましい。パケット分類エンジンは、開始信号１１２を使用して、アプリケーションエンジンの第１プログラム可能サブエンジンを開始することによって、パイプライン化パケット処理を開始することが好ましい。開始信号１１２は、アプリケーションエンジン１０６で実行する１つまたは複数のプログラムの識別を含むことが可能である。また、開始信号１１２は、パケット分類情報を含むことが可能である。アプリケーションエンジンのプログラム可能サブエンジンは、インタフェース１１４を介して、ヘッダデータと、ヘッダデータキャッシュに格納されている抽出フィールドへの直接アクセスを有することが好ましい。

【００２６】アプリケーションエンジンは、プログラム可能サブエンジンが実行しない他の処理段階を含むことが可能であるが、意思決定段階はプログラム可能サブエンジンによって実行し、柔軟性を増大することが好ましい。他の実施形態では、アプリケーションエンジンは、他の処理構造を含むことが可能である。

【００２７】また、アプリケーションデータ１１６に含まれている配置決定は、規制エンジン１２０に提供されることが好ましい。規制エンジン１２０は、また、１つまたは複数の規制ＩＤ１２４を受信することが好ましい。規制エンジン１２０は、配置決定と規制ＩＤを使用して、１つまたは複数の規制勧告１２２を生成することが好ましい。規制勧告は、配置勧告のタイプとすることが可能であり、規制結果と呼ぶことも可能である。規制勧告をアプリケーションエンジン１０６に提供し、他の配置勧告と共に使用して、配置決定を含むことが可能であるアプリケーションデータを生成することが好ましい。

【００２８】ＩＩ．プログラム可能配置論理

図４は、プログラム可能配置論理を有するパケット交換コントローラ１３０のブロック図である。パケット交換コントローラ１３０は、例えば図３のパケット交換コントローラ１００と同様とすることが可能である。パケット交換コントローラは、パケットバッファ１３２、パケット分類エンジン１３４、パターン整合探索論理１３６、アプリケーションエンジン１３８、および規制エンジン１６６を含む。

【００２９】アプリケーションエンジンは、ソース探索エンジン１４０、宛先探索エンジン１４２、および配置エンジン１４４を含む。パケット分類エンジン、ソース探索エンジン、宛先探索エンジン、および配置エンジン

は、１つまたは複数のアプリケーションプログラムを用いてプログラム可能であることが好ましい。すなわち、パケット分類エンジンとアプリケーションエンジンのサブエンジンは、それぞれ、プログラム可能マイクロコード駆動埋込み型処理エンジンを含むことが好ましい。他の実施形態では、これらのエンジンの１つまたは複数を、ハードウェアで、すなわちハードワイヤード論理として実施することが可能である。規制エンジン１６６は、ハードワイヤード論理またはプログラム可能マイクロコード駆動埋込み型処理エンジンで実施することが可能である。

【００３０】パケットバッファ１３２は、インバウンドパケット１４６を受信および格納することが好ましい。パケットバッファは、インバウンドパケットまたはその一部１４８を、パケット分類エンジン１３４に提供することが好ましい。パケット分類エンジンは、その上でプログラムされているアプリケーションプログラムを使用して、パケットを分類することが好ましく、プログラム識別１５２をアプリケーションエンジン１３８に提供することが好ましい。より詳細には、プログラム識別１５２を、アプリケーションエンジンのソース探索エンジン１４０、宛先探索エンジン１４２、および配置エンジン１４４に提供することが好ましい。本発明の一実施形態では、パケット分類エンジン１３４は、決定木に基づく分類ロジックを含む。

【００３１】プログラム識別１５２を使用して、ソース探索エンジン、宛先探索エンジン、および配置エンジンのそれぞれで実行するアプリケーションプログラムを選択することが好ましい。ソース探索エンジン、宛先探索エンジン、および配置エンジンで実行するアプリケーションプログラムは、少なくとも部分的にパケット分類情報に基づいて選択されることが好ましい。また、パケット分類情報は、プログラム識別と共に提供することが可能である。

【００３２】また、パケットバッファは、インバウンドパケットまたはその一部１５０を、パターン整合探索論理１３６に提供することが好ましい。パターン整合探索論理は、パケットまたはパケットの一部を比較するための事前定義したパターンを含むことが好ましい。例えば、パターン整合に使用するパケットの一部は、パケットヘッダデータの一部、パケットペイロードデータの一部、またはパケットヘッダデータとパケットペイロードデータの両方の一部を含むことが可能である。他の実施形態では、事前定義したパターンは、パターン整合のためにパターン整合探索論理がアクセスする、外部メモリに存在することが可能である。他の実施形態では、整合パターンは、パケット交換コントローラの動作中に変化することが可能である。

【００３３】比較を実施した後、比較の結果１５４をアプリケーションエンジン１３８に提供することが好ましい。より詳細には、比較の結果１５４を、アプリケーションエンジンの配置エンジン１４４に提供することが好ましい。一実施形態では、整合がある場合にのみ、結果を配置エンジンに提供することが可能である。

【００３４】ソース探索エンジン１４０は、インバウンドパケットのソースアドレスを使用して、ソースアドレス探索を少なくとも部分的に実施することによって、インバウンドパケットに対する配置勧告１６０を生成することが好ましい。また、配置勧告１６０は、パケット分類エンジンによって提供されたプログラム識別に従ってソース探索エンジン１４０で実行されたアプリケーションプログラムに依存することが好ましい。配置勧告１６０は、インバウンドパケットに対するセキュリティ勧告を含むことが好ましい。

【００３５】他の実施形態では、ソース探索エンジン１４０を使用して、１つまたは複数のキーをビルドすることが可能であり、次いでこれを使用して、アドレステーブルでインバウンドパケットに対するソースアドレス（ＩＰＳＡなど）を探索することが可能である。キーは、仮想ＬＡＮ識別（ＶＬＡＮ　ＩＤ）、アプリケーション識別（ＡＰＰ　ＩＤ）、およびＩＰＳＡの１つまたは複数を含むことが可能であるが、これに限定されるものではない。また、ソース探索エンジン１４０によってビルドされた１つまたは複数のキーを使用して、例えばセキュリティ勧告などの配置勧告を策定することも可能である。

【００３６】宛先探索エンジン１４２は、ソース探索エンジン１４０から出力１５６を受信することが好ましい。出力１５６は、ソースアドレスを探索するために使用するキーおよび／または探索の結果を含むことが可能である。宛先探索エンジンは、パケット分類エンジン１３４によって識別されたアプリケーションプログラムを実行して、１つまたは複数の規制識別子（ＩＤ）１６８を生成することが好ましい。規制ＩＤ１６８は、インバウンドパケットの宛先アドレスを使用する宛先アドレス探索に少なくとも部分的に基づくことができる。

【００３７】規制エンジン１６６は、規制ＩＤ１６８をキーとして使用して、規制データテーブルの規制データにアクセスすることが好ましい。規制エンジン１６６は、アクセスした規制データを使用して、１つまたは複数の規制勧告１７０を生成することが好ましい。配置エンジンが規制勧告並びに他の配置勧告を使用することにより、配置決定を含むことが可能であるアプリケーションデータを生成することが好ましい。パターン整合探索論理１３６が整合を見つけるとき、パターン整合結果１５４は、規制勧告に優先することが好ましい。規制勧告を使用して、最悪の場合の規制勧告を選択することによって、１つの勧告を生成することが好ましい。また、規制エンジンは、会計（ａｃｃｏｕｎｔｉｎｇ）機能を実施することも可能である。

【００３８】他の実施形態では、宛先探索エンジン１４２を使用して、１つまたは複数のキーをビルドすることが可能であり、次いでこれを使用して、アドレステーブルにおいて、インバウンドパケットの宛先アドレス（ＩＰＤＡなど）を探索することが可能である。キーは、仮想ＬＡＮ識別（ＶＬＡＮ　ＩＤ）、アプリケーション識別（ＡＰＰ　ＩＤ）、およびＩＰＤＡの１つまたは複数を含むことが可能であるが、これに限定されるものではない。

【００３９】配置エンジン１４４は、配置勧告１６０のセキュリティ勧告、規制勧告１７０、およびパターン整合結果１５４を含むが、これに限定されない、いくつかの配置勧告を受信することが好ましい。配置エンジンは、配置勧告、並びにパケット分類および／またはプログラム識別に基づいて、配置決定１６２を生成することが好ましい。配置決定１６２は、配置勧告の１つを含むことが可能である。一般に、パターン整合結果１５４は、規制勧告１７０に優先することが可能であり、規制勧告は、配置勧告１６０のセキュリティ勧告に優先することが可能である。配置決定１６２は、会計データ、経路指定データ、および規制データの１つまたは複数を含むことが可能であるが、これに限定されない、アプリケーションデータの一部である可能性がある。

【００４０】配置決定をパケットバッファに提供し、アウトバウンドパケット１６４として提供されるインバウンドパケットの編集に使用することが好ましい。また、規制および会計のために、配置決定を再度規制エンジンに供給することが好ましい。例えば、インバウンドパケットがドロップされるとき、規制エンジンは、そのことを認識するべきである。他の実施形態では、宛先探索エンジンは、規制エンジンを含むことが可能である。そのような場合、配置決定は、規制および会計のために、宛先探索エンジンに提供されることが好ましい。

【００４１】図５は、複数の配置勧告と分類情報を使用して、配置決定をプログラムで生成するプロセスの流れ図である。ステップ１８０で、例えば、図４のパケットバッファ１３２などのパケットバッファは、インバウンドパケットを受信することが好ましい。パケットバッファでは、パケットヘッダデータを抽出して、ヘッダデータキャッシュに格納することが可能である。

【００４２】インバウンドパケットまたはインバウンドパケットの一部は、ヘッダデータを含むことが可能であり、例えば図４のパターン整合探索論理１３６などのパターン整合探索論理に提供されることが好ましい。ステップ１８２で、パターン整合探索論理は、インバウンドパケットまたはインバウンドパケットの一部と、所定のパターンとの間のパターン整合探索を実施して、ステップ１８８で示されているように、パターン整合勧告を生成することが好ましい。所定のパターンは、例えば、内部メモリまたは外部メモリに含むことが可能である。他の実施形態では、整合パターンは、動的に変化する可能性がある。

【００４３】一方、インバウンドパケットまたはその一部を、例えば、図４のパケット分類エンジン１３４など、パケット分類エンジンに提供することも好ましい。ステップ１８４で、パケット分類エンジンは、パケットを分類することが好ましく、パケットの分類に基づいて、アプリケーションプログラムを識別することが好ましい。ステップ１８６で、プログラム識別を、例えば、図４のアプリケーションエンジン１３８など、アプリケーションエンジンのソース探索エンジン、宛先探索エンジン、および配置エンジンに提供することが好ましい。プログラム識別は、これらのサブエンジンで実行するアプリケーションプログラムを示すことが好ましい。また、パケット分類情報を、ソース探索エンジン、宛先探索エンジン、および配置エンジンに提供することが好ましい。ソース探索エンジンは、ステップ１９０でセキュリティ勧告を生成することが好ましく、一方規制エンジンは、宛先探索エンジンからの規制ＩＤを使用して、ステップ１９２で規制勧告を生成することが好ましい。

【００４４】ステップ１９４で、パターン整合勧告、セキュリティ勧告、および規制勧告を配置エンジンに提供することが好ましい。配置エンジンは、選択したアプリケーションプログラムと配置勧告の１つまたは複数を使用して、配置決定を生成することが好ましい。配置決定をパケットバッファに提供し、これを使用して、ステップ１９６で、インバウンドパケットを編集して、アウトバウンドパケットとして送信することが好ましい。また、例えば、規制および会計などのために、ステップ１９８で、配置決定を再度規制エンジンに供給することが好ましい。

【００４５】ＩＩＩ．マルチレベル規制

本発明の一実施形態では、規制エンジンは、パケット交換コントローラを通って進むトラフィックを複数のポリシーグループに基づいて規制するマルチレベル規制論理を使用することが好ましい。顧客は、自分の帯域幅契約において、適切なポリシーグループとそれらのグループに適用可能な帯域幅とを指定することが好ましい。例示的なシナリオでは、顧客は、自分の帯域幅契約において、特定のポートについて、１Ｇｂｐｓのデータトラフィックに対し支払うと指定することが可能である。さらに、顧客は、異なるデータフロー限度を自分のカンパニのサブネットに割り当てることが可能である。例えば、顧客は、エンジニアリングサブネットを３００Ｍｂｐｓに限定し、会計サブネットを１００Ｍｂｐｓに限定することが可能である。さらに、顧客は、カンパニ全体で、ウェブのトラフィックを２００Ｍｂｐｓに限定すると指定することが可能である。したがって、トラフィックのタイプに関係なく、ポート毎にトラフィックのみを規制する代わりに、ウェブトラフィックと、エンジニアリン

グサブネットまたは会計サブネットを発信源とするトラフィックを、それぞれの閾値に基づいて、識別および規制することが可能である。

【００４６】さらに、サービスプロバイダと顧客との間の帯域幅契約により、ＱｏＳ動作を判定することも可能である。ＱｏＳ動作は、フロー条件を満たすトラフィックに適用可能なＱｏＳを識別することが可能である。ＱｏＳ動作は、最大帯域幅、最小帯域幅、ピーク帯域幅、優先順位、待ち時間、ジッタ、最大キュー深度、最大キューバッファなどを示すことが可能である。

【００４７】帯域幅規制機能は、一般的な解決法の一部として、フロー毎に進入データ速度を制御して、トラフィックのフローを規制するなど限定し、成形することが好ましい。図６は、異なるフローの規制を示すブロック図である。規制パラメータは、時間毎にバイト単位でコミッテッド情報速度（ＣＩＲ）を定義し、並びにコミッテッドバーストサイズ（ＣＢＳ）と余剰バーストサイズ（ＥＢＳ）をどちらもバイト単位で定義することによって確立することが好ましい。バケットは、第１バケット（ドロップ適格（ＤＥ）バケット）２００と第２バケット（ドロップバケット）２０２に分類、すなわちマークされることが好ましい。

【００４８】バケットが所与の進入速度で提示されるとき、各バケット内の現在のバランスと、ＣＢＳおよびＥＢＳに対する関係とによって、マークされることが好ましい。第１バケットは、廃棄適格（ＤＥ）バランスを維持することが好ましい。第２バケットは、ドロップバランスを維持することが好ましい。進入速度がＣＢＳより小さい場合、バケットに転送とマークすることが好ましい。進入速度がＣＢＳより大きいかまたはそれに等しいが、ＥＢＳより小さい場合、バケットにＤＥとマークすることが好ましい。進入速度がＥＢＳより大きいかまたはそれに等しい場合、バケットにドロップとマークすることが好ましい。

【００４９】図７は、本発明の一実施形態で、複数のポリシーレベルに基づいてデータバケットを規制するために使用する規制データテーブル２５０である。規制データテーブル２５０は、図４の規制エンジン１６６と同様とすることが可能である、規制エンジンに格納することが可能である。また、規制データテーブル２５０は、規制データベースと呼ぶことも可能である。

【００５０】規制データテーブル２５０は、例えば、図４のバケット交換コントローラ１３０などバケット交換コントローラを通って進むトラフィックの現在速度をチェックする規制データを含む。規制データテーブル２５０は、多様な方式で構成することが可能であるが、順次エントリとして構成し、各エントリが、特定のポリシーグループに関連付けられている規制データ２５２を提供することが好ましい。各規制データ２５２は、独自の規制識別子（ＩＤ）／キー２５４によって識別することが好ましい。

【００５１】規制ＩＤ２５４は、バケットを分類することが可能である異なるポリシーグループを識別することが好ましい。各規制ＩＤ２５４は、顧客識別子２５４ａおよび／またはアプリケーション識別子２５４ｂから構成されることが好ましい。顧客識別子は、ソースアドレス、物理ポートなどに基づいて、特定の顧客を識別することが好ましい。アプリケーション識別子２５４ｂは、バケットに関連付けられているアプリケーションのタイプに基づいて、アプリケーションＲＡＭによって割り当てられた内部識別子であることが好ましい。例示的なアプリケーションは、ウェブアプリケーション、ボイスオーバーＩＰ（ＶｏＩＰ）アプリケーションなどを含む。

【００５２】次の規制ＩＤ２５６により、規制データベースの入れ子式探索が、バケットに適用可能な追加のポリシーグループを識別することが可能になることが好ましい。また、それらのポリシーグループに関連付けられている規制データ２５２を検索して、現在のバケットの速度チェックを実施することが好ましい。

【００５３】各規制データ２５２は、現在の帯域幅、並びに規制ＩＤ２５４によって識別された各ポリシーグループの帯域幅の限度を示すことが好ましい。ドロップバランス２５２ｃとドロップ適格（ＤＥ）バランス２５２ｄは、バケット交換コントローラを通って進むトラフィックの量のカウントを維持することが好ましい。ドロップバランス２５２ｃとＤＥバランス２５２ｄを、それぞれドロップ限度２５２ｅおよびＤＥ限度２５２ｆと比較して、現在のバケットを転送する、ＤＥとマークをして転送する、または即座にドロップすることを勧告することが好ましい。ドロップバランス２５２ｃは、ＤＥバランス２５２ｄがＤＥ限度２５２ｆより大きくなるまで、インクリメントされないことが好ましい。

【００５４】各規制データ２５２は、さらに、最後のバランス計算が実施された時間を示すタイムスタンプ２５２ｂを含むことが好ましい。現在の時間とタイムスタンプの情報が与えられれば、最後のバランス計算から経過した時間を測定して、この時間中のトラフィック速度を計算することが可能である。タイムスタンプの増分のサイズは、やはり規制データテーブル２５０に維持されているバジェット（ＣＩＲ）２５２ａの値に基づいて、調整することが可能である。例えば、バジェット値は、本発明の一実施形態では、タイムスタンプ増分あたりのバイト数として定義することが可能である。

【００５５】示した規制データテーブル２５０では、規制エンジンは、第１規制ＩＤに基づいて、速度チェック２５８または２６０を実施して、勧告したバケットの配置を示す第１ポリシー結果を生成することが好ましい。さらに、規制エンジンは、バケットが追加のポリシーグループに基づいて規制されるかを判定することが好ましい。そのために、ポリシーエンジンは、次の規制ＩＤフ

ィールド２５６を調査して、ＩＤによって識別された規制データを検索することが好ましい。次いで、第２速度チェック２６２を同じバケットに対して実施して、第２速度チェックに基づいて、第２ポリシー結果を生成することが好ましい。追加の速度チェックを続行して、次のポリシーＩＤフィールド２５６に関する値に基づいて実施することが可能である。本発明の一実施形態では、各バケット対し最高で４つの規制アルゴリズムを実行しながら、回線速度の性能を維持することが可能である。他の実施形態では、４つより多いまたは少ない規制アルゴリズムを実行することが可能である。

【００５６】図８は、マルチレベル規制プロセスの例示的な流れ図である。プロセスは、ステップ３００で開始し、規制エンジンは、入来バケットに対する新しい規制ＩＤを受信することが好ましい。ステップ３０２で、規制エンジンは、規制ＩＤに関連付けられている規制データを検索することが好ましい。ステップ３０４で、規制エンジンは、好ましくは以下の式により、新しいドロップバランスまたはＤＥバランスを計算することが好ましい。

【００５７】バランス$_{new}$＝バランス$_{old}$－［バジェット×（時間－タイムスタンプ）］＋バケットサイズ上式で、バランス$_{new}$とバランス$_{old}$は、それぞれ、規制ＩＤに関連付けられているドロップバケットまたはＤＥバケットに対する新しいバランスと現在のバランスを表すことが好ましい。バジェットは、ＣＩＲなど、規制ＩＤに関連付けられているバジェット２５２ａを表すことが好ましい。現在のドロップバランスとＤＥバランスは、それぞれ、ドロップバランス２５２ｃとＤＥバランス２５２ｄに対応する。時間とタイムスタンプは、それぞれ、規制ＩＤに関連付けられている現在の時間とタイムスタンプ２５２ｂを表すことが好ましい。バケットサイズは、処理しているバケットのサイズを表すことが好ましい。

【００５８】ステップ３０６で、新しいドロップバランスまたはＤＥバランスを、ドロップ限度２５２ｅまたはＤＥ限度２５２ｆに対して適用する。ＤＥ限度を超えるまで、このバランスをＤＥバランスに対して適用することが好ましい。規制エンジンは、ＤＥバランスとＤＥ限度を比較することが好ましく、ＤＥバランスがＤＥ限度より小さい場合、バケットを転送する決定をすることが好ましい。ＤＥバランスがＤＥ限度を超える場合、このバランスをドロップバランスに適用することが好ましい。次いで、規制エンジンは、ドロップバランスとドロップ限度を比較することが好ましく、ドロップバランスがドロップ限度より小さい場合、ＤＥとマークして、バケットを転送することを決定することが好ましい。しかし、ドロップ限度を超えた場合、規制エンジンは、バケットを即座に廃棄することを決定することが好ましい。

【００５９】例えば、実際には、新しいバランスを計算し、次いで、ＤＥ限度とドロップ限度を比較して、転送ステータスを決定することが好ましい。バランスは、転送結果に基づいて更新されることが好ましい。例えば、バケットに転送とマークされている場合、ＤＥバランスを更新することが好ましい。すなわち、バケットに転送とマークされているとき、例えば、図６の第１バケット２００などＤＥバケットは、満たされていることが好ましい。他の例として、バケットにＤＥとマークされている場合、ドロップバランスを更新することが好ましい。すなわち、バケットにＤＥとマークされているとき、図６の第２バケット２０２などドロップバケットは、満たされている。この時点で、ＤＥバケットはすでに満ちている。他の例として、バケットにドロップとマークされているとき、両方のバケットともこの時点で満ちているので、ＤＥバランスもドロップバランスも更新されない。

【００６０】ステップ３０８で、現在のバケットに対して示された追加の規制ＩＤが存在するかについて判定する。存在する場合、プロセスはステップ３０２に戻って、追加の規制ＩＤによって識別された規制データを検索し、追加のポリシー結果を生成する。

【００６１】ステップ３１０で、規制エンジンが、例えば、図４の配置エンジン１４４など、配置エンジンに規制結果を通知することが好ましく、これはまた、規制勧告と呼ぶことが可能である。複数のポリシー結果が、処理しているバケットに利用可能である場合では、規制エンジンは、最も保守的な規制結果、すなわち最悪の場合の規制結果を選択することが好ましく、その結果を配置エンジンに戻すことが好ましい。配置エンジンは、規制結果、セキュリティ勧告などの他の配置勧告、およびパターン整合結果を使用して、配置決定を生成することが好ましい。

【００６２】ステップ３１２で、規制エンジンは、配置エンジンから、配置決定の通知を受信することが好ましい。配置決定は、バケットを転送したか、ＤＥとマークして転送したか、またはドロップしたかに関する決定を含むことが可能である。ステップ３１４で、規制エンジンは、バケットを転送したかを判定することが好ましい。そうである場合、転送したバケットに関連付けられている各規制データは、増大したトラフィックを反映して、ステップ３１６で更新される。

【００６３】規制データベースで更新される値は、ＤＥバランス、ドロップバランス、およびタイムスタンプの１つまたは複数を含むことが好ましい。ＤＥバランスは、ＤＥ限度より小さい場合に更新されることが好ましい。ドロップバランスは、ＤＥバランスがＤＥ限度より大きく、ドロップバランスがドロップ限度より小さい場合に更新されることが好ましい。両方のバランスが、それぞれの限度を超えている場合、どちらも更新されないことが好ましい。例えば、あらゆる場合に、フレームな

どパケットが、配置決定によって示された任意の理由でドロップされる場合、「バケットサイズ」（バケットのサイズ）の値をどちらのバランスにも追加しないことが望ましい。このようにして、交換ファブリックに入来するパケットについて、正確なカウントが作成されることが好ましい。

【００６４】ＩＶ．据置きデビッティングを備えるフロー速度規制

本発明の一実施形態では、据置き（ｄｅｆｅｒｒｅｄ）デビッティングを、フロー速度規制と共に使用することが好ましい。図９は、本発明のこの実施形態における、据置きデビッティングを備え、フロー速度規制を有するパケット交換コントローラのブロック図４００である。据置きデビッティングは、マルチレベル規制論理と併用することが可能である。

【００６５】図９に示すように、フィールド抽出装置４０２は、パケットを受信し、フロー情報を一般決定論理４０８と据置きデビット規制論理４１０に提供し、パケットをパケットサイズ計算装置４０４に提供する。パケットサイズ計算装置４０４は、出力をパケットサイズバッファ４０６に提供し、パケットをパケットバッファ４１２に提供する。一般決定論理４０８と据置きデビット規制論理４１０は、それぞれ、一般決定結果と規制結果を配置論理４１４に提供する。配置論理は、配置結果をパケットバッファ４１２に提供する。また、配置論理４１４は、配置結果を据置きデビット規制論理４１０に提供し、これは、据置きデビッティングのために、配置結果とパケットサイズ情報を使用する。

【００６６】フロー速度規制は、異なるサービス品質を受ける資格のある顧客が、共有ネットワークの帯域幅を競い合っているので、データ通信ネットワーキングではますます重要になってきている。通常、フロー速度規制は、フロー内のパケットを、フローについて定義された１つまたは複数の帯域幅契約と比較して、（ｉ）条件なしでパケットを承認する；（ｉｉ）条件付きでパケットを承認する（例えば、パケットに廃棄適格とマークする）；または（ｉｉｉ）パケットを廃棄する、について解決することを含む。

【００６７】通常、フロー速度規制方式は、「トークンバケット」を維持して、各帯域幅契約の下で現在利用可能な帯域幅を表す。通常、契約用に維持しているバケットに、現在十分なトークンが存在する場合、バケットは、フローの帯域幅契約内にあると見なされ、契約用に維持しているバケットに、現在十分なトークンが存在しない場合、バケットは、契約を超過していると見なされる。時間が経過するにつれ、時間クレジットを介してバケットにトークンが追加される。バケットサイズデビットを介してバケットが承認される際に、バケットからトークンが減算される。

【００６８】トークンバケットの状態を維持するために使用する一般的な式は、以下の通りである。

【００６９】$TC_{new} = TC_{old} + C - D$

上式で、

TC_{new}＝新しいトークンカウント

TC_{old}＝古いトークンカウント

Ｃ＝時間クレジット

Ｄ＝サイズデビット

である。

【００７０】トークンバケット状態式の１つのインスタンスを適用して、以下のように、簡単な承認／廃棄規制決定を実施することが可能である。フロー内のパケットが規制決定のために到着するとき、先行パケットに関する規制決定から経過した時間を反映している時間クレジットＣを追加し、現在のパケットのサイズを反映しているサイズデビットＤを減算することによって、フローの帯域幅契約に対する新しいトークンカウントTC_{new}を計算する。次いで、フローの帯域幅契約に対する新しいトークンカウントTC_{new}を、ゼロと比較する。新しいトークンカウントTC_{new}がゼロより大きいまたはゼロに等しい場合、現在のパケットは、帯域幅契約内にあり、承認される。新しいトークンカウントTC_{new}がゼロより小さい場合、現在のパケットは、帯域幅契約を超過しており、廃棄される。

【００７１】トークンバケット状態式の２つのインスタンスを同じフローに適用して、より洗練された規制結果を提供することが可能である。例えば、廃棄トークンバケットおよび廃棄適格トークンバケットを、フローに対して別々に維持することが可能である。その場合、新しい廃棄トークンカウントTC_{new-de}がゼロより大きいかまたはゼロに等しいが、新しい廃棄トークンカウントTC_{new-d}がゼロより小さい場合、現在のパケットは、廃棄帯域幅契約内にあるが、廃棄適格帯域幅契約を超過している。したがって、現在のパケットは、（廃棄適格帯域幅契約を超過しているので）廃棄適格とマークされることを条件として、（ドロップ帯域幅契約内にあるので）承認される。そのような３レベル「２重トークンバケット」規制方式は、「Ａ Ｓｉｎｇｌｅ Ｒａｔｅ Ｔｈｒｅｅ ＣｏｌｏｒＭａｒｋｅｒ」という名称のＩＥＴＦ Ｒｅｑｕｅｓｔ ｆｏｒ Ｃｏｍｍｅｎｔ ２６９７に記載されている。

【００７２】現況技術のパケット交換コントローラで、トークンバケット状態式を適用して、高速データフローを規制することは、特に、現在のパケットのサイズを反映しているサイズデビットＤを減算しその後規制決定を行うという教示に関して、実用上の困難に直面していた。第１に、現在のパケットのサイズは、規制論理の外部で決定することが可能である。したがって、現在のパケットに対するサイズデビットＤは、規制決定が行われるときには利用可能でない可能性がある。第２に、規制決定のみでは、パケットの最終配置を指図しない可能性

がある。したがって、現在のパケットに対するサイズデビットＤの差引き（ｄｅｄｕｃｔｉｏｎ）は、後に逆のことを要求する可能性がある。第３に、現在のパケットに対するサイズデビットＤは、差引き後規制決定を行う場合、（全てではないが）ほとんどのパケットを収容するのに十分なトークンがバケットに存在する場合でも、現在のパケットが帯域幅契約を超過するとみなされることになる。

【００７３】一方、現在のパケットに対するサイズデビットＤを差し引きその後規制決定を行う実用的な利点は、高速コントローラではデータの転送速度は最大バケットサイズより指数関数的に大きいので、明らかではない。サイズデビットＤが後に適度な時間内で作成される限り、せいぜいフローに対する名目上のおよび一時的な帯域幅契約違反が行われる程度である。

【００７４】本発明のこの実施形態では、据置きデビッティングを使用して、一般的なトークンバケット状態式を適用して高速データフローを規制することに関する上記の困難を克服することが好ましい。

【００７５】例えば、データ規制方法を提供することが可能である。データ規制方法は、パケットを受信する；時間クレジットを第１トークンカウントに追加して第２トークンカウントを生成する；第２トークンカウントを適用してパケットに対する規制結果を生成する；パケットに対する規制結果を適用し第２トークンカウントからサイズデビットを減算して第３トークンカウントを生成する、または第２トークンカウントからサイズデビットを減算せず第３トークンカウントを生成しない、ことを含むことが好ましい。

【００７６】さらに、データ規制方法は、第２パケットを受信する；時間クレジットを第２トークンカウントに追加して第４トークンカウントを生成する；第４トークンカウントを適用して第２パケットに対する規制結果を生成する、ことを備えることが可能である。

【００７７】また、他のデータ規制方法を提供することが可能である。このデータ規制方法は、パケットを受信する；時間クレジットを第１トークンカウントに追加して第２トークンカウントを生成する；第２トークンカウントを適用してパケットに対する規制結果を生成する；パケットに対する規制結果を適用してパケットに対する配置結果を生成する；パケットに対する配置結果を適用し第２トークンカウントからサイズデビットを減算して第３トークンカウントを生成する、または第２トークンカウントからサイズデビットを減算せず第３トークンカウントを生成しない、ことを含むことが好ましい。

【００７８】このデータ規制方法では、少なくとも１つの他の勧告を有する勧告として規制結果を適用して、バケットに対する配置結果を生成することが可能である。

【００７９】他のデータ規制方法はバケットを受信する；時間クレジットをトークンカウントのそれぞれに追加して第２トークンカウントのそれぞれを生成する；第２トークンカウントのそれぞれを適用してバケットに対する規制結果を生成する；バケットに対する規制結果を適用し第２トークンカウントの少なくとも１つからサイズデビットを減算して少なくとも１つの第３トークンカウントを生成する、または第２トークンカウントの少なくとも１つからサイズデビットを減算せず少なくとも１つの第３トークンカウントを生成しない、ことを含むことが好ましい。

【００８０】他のデータ規制方法は、バケットを受信する；時間クレジットをトークンカウントのそれぞれに追加して第２トークンカウントのそれぞれを生成する；第２トークンカウントのそれぞれを適用してバケットに対する規制結果を生成する；バケットに対する規制結果を適用してバケットに対する配置結果を生成する；バケットに対する配置結果を適用し第２トークンカウントの少なくとも１つからサイズデビットを減算して少なくとも１つの第３トークンカウントを生成する、または第２トークンカウントの少なくとも１つからサイズデビットを減算せず少なくとも１つの第３トークンカウントを生成しない、ことを含むことが好ましい。

【００８１】以下のデータ規制方法は、さらに、本発明の一実施形態における、据置きデビッティングを備えるフロー速度規制を示す。

【００８２】データ規制方法は、パケットを受信する；時間クレジットを第１トークンカウントに追加して、第２トークンカウントを生成する；第２トークンカウントを適用してバケットに対する規制結果を生成する；規制結果を適用し、第２トークンカウントからサイズデビットを減算して第３トークンカウントを生成する、または第２トークンカウントからサイズデビットを減算せず第３トークンカウントを生成しない、ことを含むことが好ましい。

【００８３】このデータ規制方法は、さらに、第２パケットを受信する；時間クレジットを第２トークンカウントに追加して第４トークンカウントを生成する；第４トークンカウントを適用して第２パケットに対する規制結果を生成する、ことを含むことが好ましい。

【００８４】他のデータ規制方法は、バケットを受信する；時間クレジットを第１トークンカウントに追加して第２トークンカウントを生成する；第２トークンカウントを適用してバケットに対する規制結果を生成する；規制結果を適用してバケットに対する配置結果を生成する；配置結果を適用し第２トークンカウントからサイズデビットを減算して第３トークンカウントを生成する、または第２トークンカウントからサイズデビットを減算せず第３トークンカウントを生成しない、ことを含むことが好ましい。少なくとも１つの他の勧告を有する勧告として規制結果を適用して配置結果を生成することが可能である。

【００８５】他のデータ規制方法は、パケットを受信する；時間クレジットをトークンカウントのそれぞれに追加して第２トークンカウントのそれぞれを生成する；第２トークンカウントのそれぞれを適用してパケットに対する規制結果を生成する；規制結果を適用し第２トークンカウントの少なくとも１つからサイズデビットを減算して少なくとも１つの第３トークンカウントを生成する、または第２トークンカウントの少なくとも１つからサイズデビットを減算せず少なくとも１つの第３トークンカウントを生成しない、ことを含むことが好ましい。

【００８６】他のデータ規制方法は、パケットを受信する；時間クレジットをトークンカウントのそれぞれに追加して第２トークンカウントのそれぞれを生成する；第２トークンカウントのそれぞれを適用してパケットに対する規制結果を生成する；規制結果を適用してパケットに対する配置結果を生成する；配置結果を適用し第２トークンカウントの少なくとも１つからサイズデビットを減算して少なくとも１つの第３トークンカウントを生成する、または第２トークンカウントの少なくとも１つからサイズデビットを減算せず少なくとも１つの第３トークンカウントを生成しない、ことを含むことが好ましい。

【００８７】ある特定の実施形態に関して、本発明について説明したが、当業者なら、本発明の範囲および精神から決して逸脱せずに、変形形態を工夫することに何ら困難はないであろう。したがって、本発明は、詳細に説明した以外の方式で実行することが可能であることを理解されたい。したがって、本発明の本実施形態は、あらゆる点で例示的であって限定的ではなく、本発明の範囲は、上記の説明ではなく、添付の請求項とその同等物によって示されていると見なすべきである。

【図面の簡単な説明】

【図１】本発明の一実施形態を使用する、パケット交換ノードを含むネットワーク環境の図である。

【図２】本発明の一実施形態における、交換インタフェースのブロック図である。

【図３】本発明の一実施形態における、プログラム可能パケット交換コントローラのブロック図である。

【図４】本発明の一実施形態における、プログラム可能配置論理を有するパケット交換コントローラのブロック図である。

【図５】本発明の一実施形態における、複数の配置勧告と分類情報を使用して配置決定をプログラムで生成するプロセスの流れ図である。

【図６】パケットを異なる分類にマークするプロセスを示すブロック図である。

【図７】本発明の一実施形態における、複数のポリシーレベルに基づいてデータパケットを規制するために使用する規制データテーブルである。

【図８】本発明の一実施形態における、マルチレベル規制プロセスの流れ図である。

【図９】本発明の一実施形態における、据置きデビッティングを備えるフロー速度規制を有するパケット交換コントローラのブロック図である。

【符号の説明】

１００　プログラム可能パケット交換コントローラ
１０２、１３２、４１２　パケットバッファ
１０４、１３４　パケット分類エンジン
１０６、１３８　アプリケーションエンジン
１２０、１６６　規制エンジン
１２２、１７０　規制勧告
１２４　規制ＩＤ
１３０　パケット交換コントローラ
１３６　パターン整合探索論理
１４０　ソース探索エンジン
１４２　宛先探索エンジン
１４４　配置エンジン
１４６　インバウンドパケット
１４８、１５０　インバウンドパケットまたはその一部
１５２　プログラム識別
１５４　パターン整合結果
１５６　出力
１６０　配置勧告
１６２　配置決定
１６４　アウトバウンドパケット
１６８　規制識別子
２００　第１パケット（ドロップ適格（ＤＥ）パケット）
２０２　第２パケット（ドロップパケット）
２５０　規制データテーブル
２５２　規制データ
２５２ａ　バジェット（ＣＩＲ）
２５２ｂ　タイムスタンプ
２５２ｃ　ドロップバランス
２５２ｄ　ドロップ適格（ＤＥ）バランス
２５２ｅ　ドロップ限度
２５２ｆ　ＤＥ限度
２５４　規制識別子（ＩＤ）／キー
２５４ａ　顧客識別子
２５４ｂ　アプリケーション識別子
２５６　次の規制ＩＤフィールド
２５８、２６０　速度チェック
２６２　第２速度チェック
４０２　フィールド抽出装置
４０４　パケットサイズ計算装置
４０６　パケットサイズバッファ
４０８　一般決定論理
４１０　据置きデビット規制論理
４１４　配置論理

【図１】

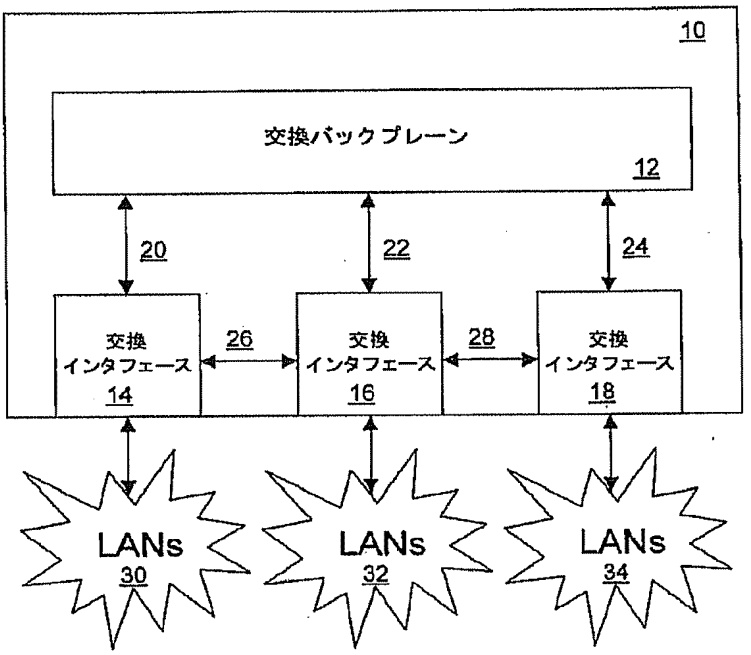

FIG. 1

【図２】

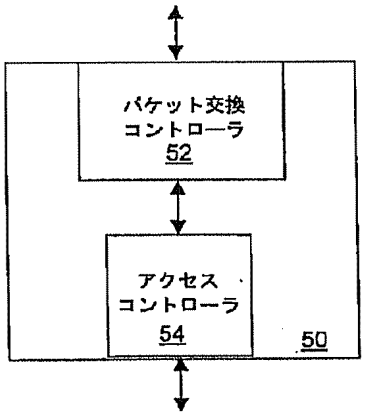

FIG. 2

【図３】

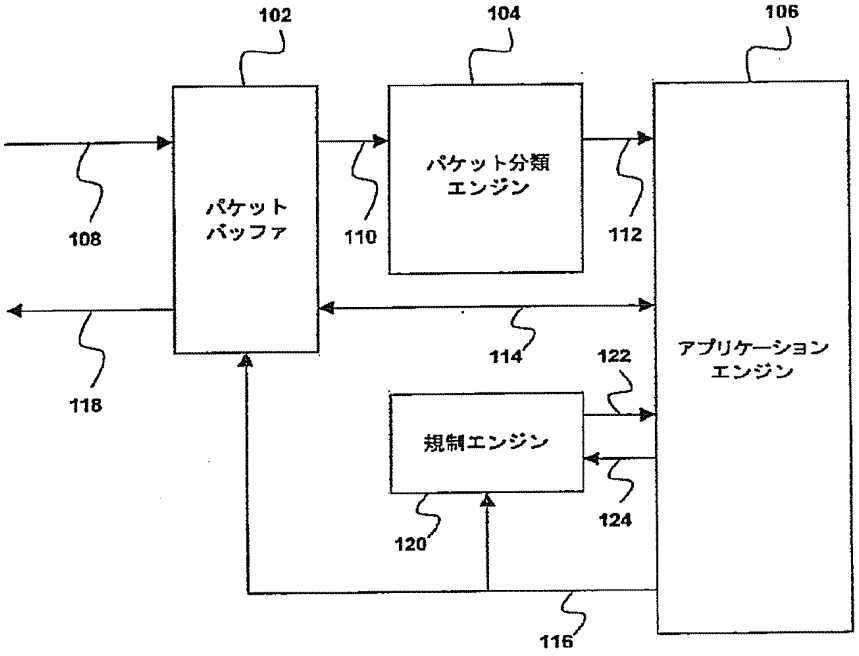
102
104
106
パケット
バッファ
パケット分類
エンジン
アプリケーション
エンジン
規制エンジン
108
110
112
114
116
118
120
122
124
100

FIG. 3

【図４】

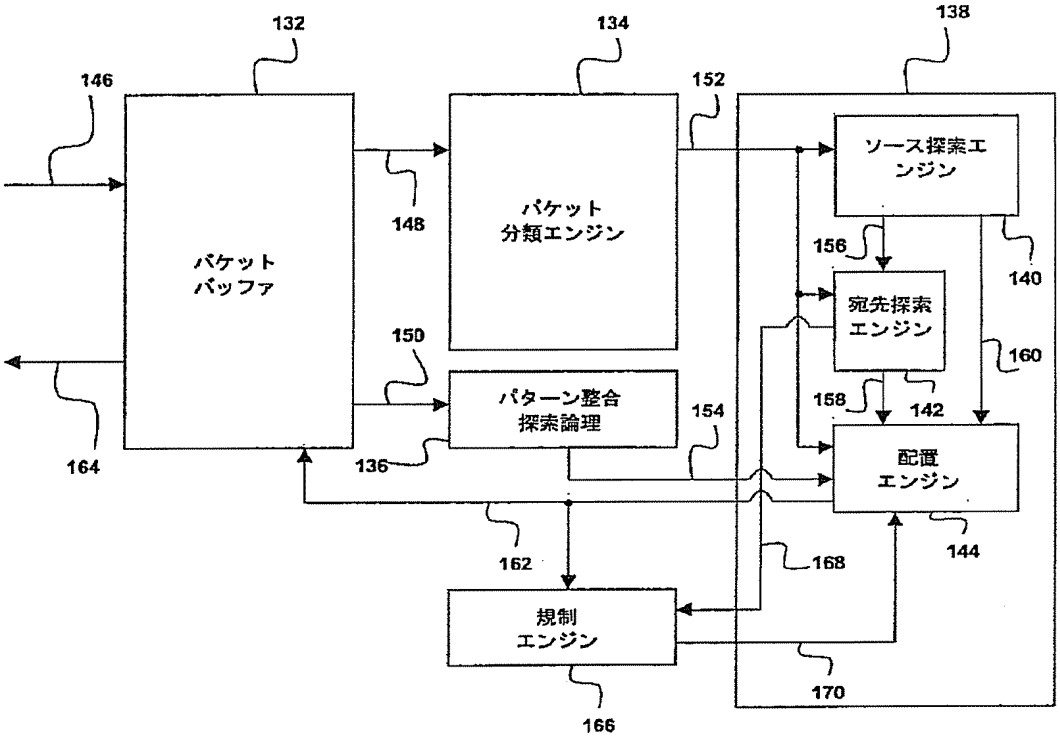
132
134
138
146
152
パケット
バッファ
148
パケット
分類エンジン
ソース探索エ
ンジン
156
140
宛先探索
エンジン
150
160
パターン整合
探索論理
158
142
154
164
136
配置
エンジン
162
144
168
規制
エンジン
170
166
130

FIG. 4

【図５】

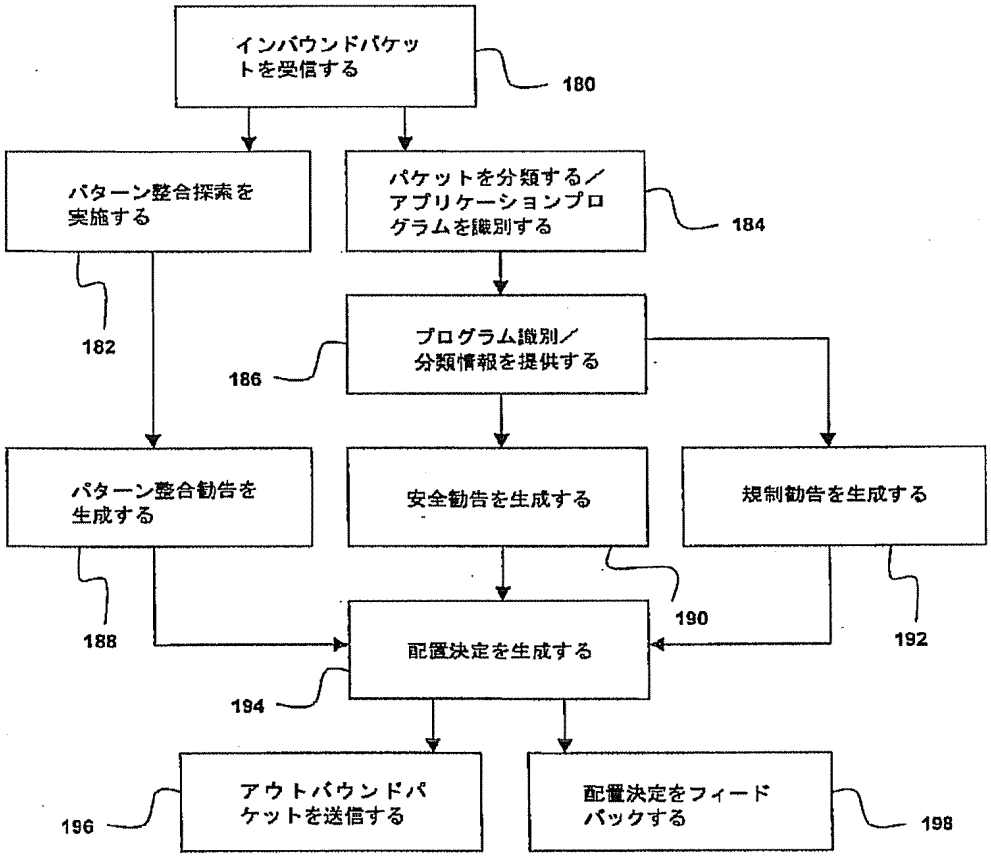
インバウンドパケットを受信する
180
パターン整合探索を実施する
パケットを分類する／アプリケーションプログラムを識別する
184
182
プログラム識別／分類情報を提供する
186
パターン整合勧告を生成する
安全勧告を生成する
規制勧告を生成する
188
190
192
配置決定を生成する
194
アウトバウンドパケットを送信する
196
配置決定をフィードバックする
198

FIG. 5

【図６】

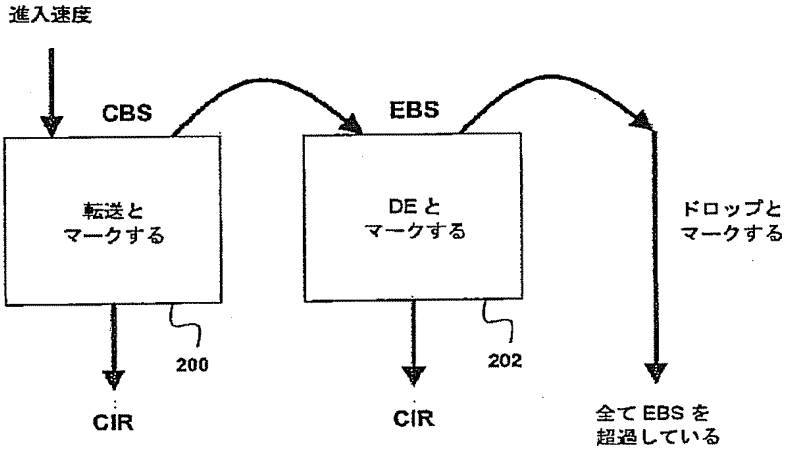
進入速度
CBS
EBS
転送と
マークする
DE と
マークする
ドロップと
マークする
200
202
CIR
CIR
全て EBS を
超過している

FIG. 6

【図７】

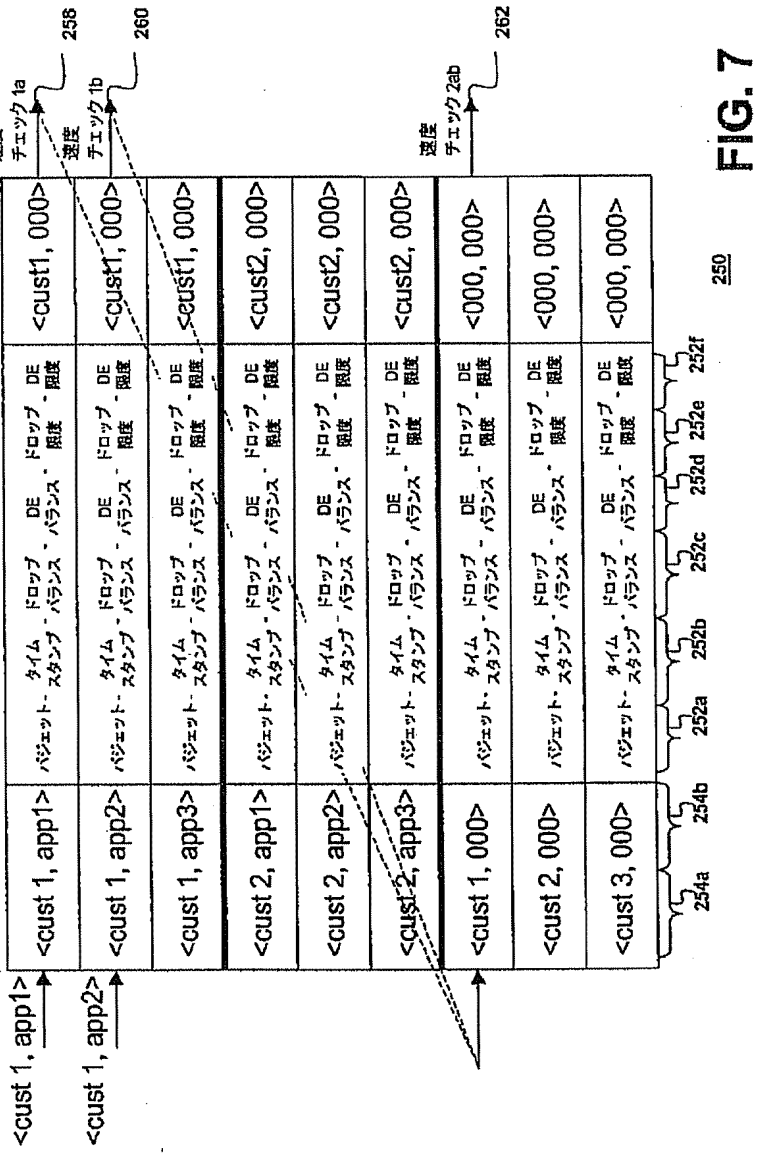
254
252
256
キー
規制データ
次のキー
速度チェック 1a
速度チェック 1b
速度チェック 2ab
258
260
262
<cust 1, app1>
<cust 1, app2>
<cust 1, app3>
<cust 2, app1>
<cust 2, app2>
<cust 2, app3>
<cust 1, 000>
<cust 2, 000>
<cust 3, 000>
バジェット・タイムスタンプ・ドロップ・バランス・DE バランス・ドロップ・DE 限度
<cust1, 000>
<cust2, 000>
<000, 000>
254a
254b
252a
252b
252c
252d
252e
252f
250

FIG. 7

【図８】

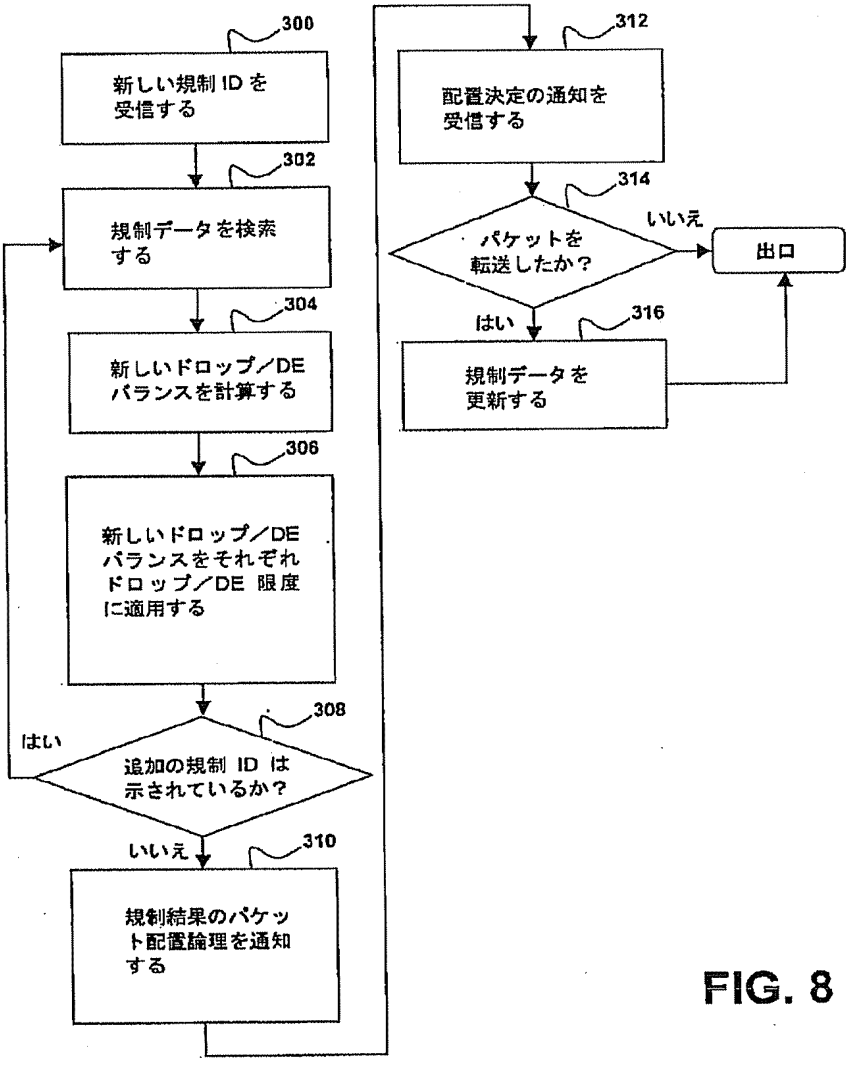

FIG. 8

* パランス$_{new}$＝パランス$_{old}$－[バジェット*(時間－タイムスタンプ)]＋パケットサイズ

【図９】

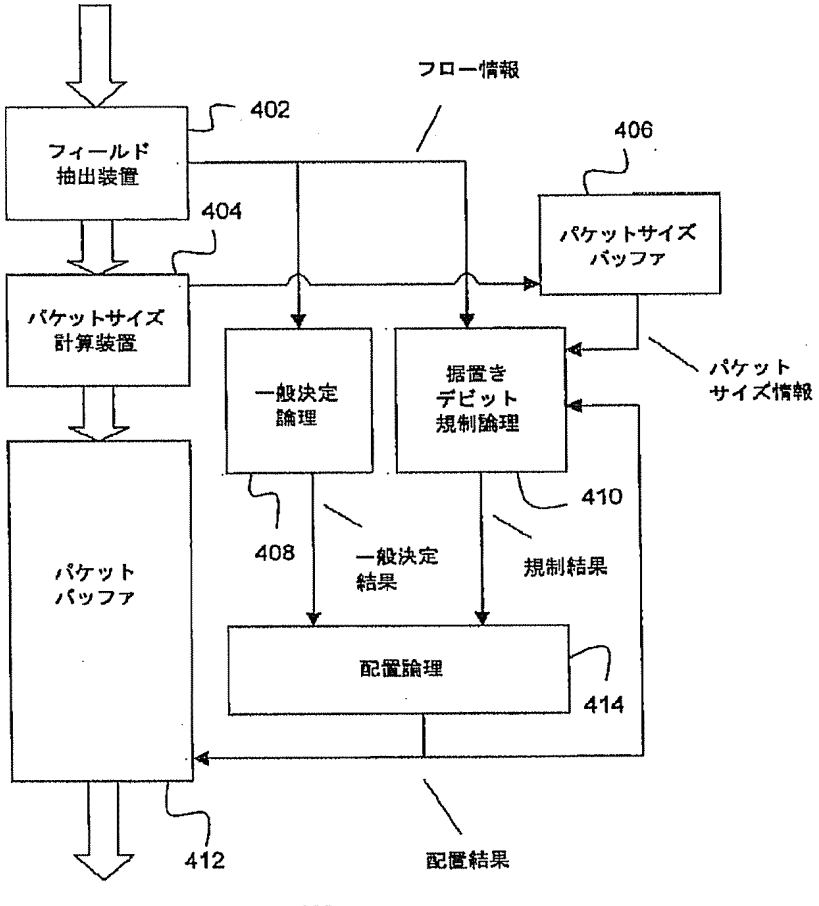

FIG. 9

フロントページの続き

(72)発明者 ケリー・フロム
アメリカ合衆国、ワシントン・99037、ベラデール、チエリル・コート・サウス・1911

(72)発明者 デニス・ボール
アメリカ合衆国、ワシントン・99019、リバテイ・レイク、ノース・オーモンド・ロード・1211

Ｆターム(参考) 5K030 GA14 HA08 HB16 KA05 LA03 LC09 LC13

[外国語明細書]

1. Title of Invention

PACKET PROCESSOR WITH MULTI-LEVEL POLICING LOGIC

2. Claims

1. A packet switching controller comprising:

an input for receiving a packet;

a policing element for classifying the packet into a plurality of policeable groups,

wherein the packet is compared against one or more bandwidth contracts defined for the policeable groups to produce one or more policing results.

2. The packet switching controller of claim 1 wherein the policing element includes a policing database, a first policeable group identifier is applied to the policing database to retrieve first policing data and a second policeable group identifier, the first policing data is applied to produce a first policing result, the second policeable group identifier is applied to the policing database to retrieve second policing data, and the second policing data is applied to produce a second policing result.

3. The packet switching controller of claim 1 further comprising a disposition engine for making a disposition decision for the packet, wherein the policing results include one or more disposition recommendations, and the disposition engine uses the policing results and at least one other disposition recommendation to make the disposition decision for the packet.

4. The packet switching controller of claim 1 wherein the policing results are combined into a single result by taking a worst case policing result.

5. A method of processing a packet using a policing element, the method comprising the steps of:

receiving the packet;

classifying the packet into a plurality of policeable groups; and

comparing the packet against one or more bandwidth contracts defined for the policeable groups to produce one or more policing results.

6. The method of processing a packet of claim 5 wherein the policing element includes a policing database, and the method further comprises the steps of:

applying a first policeable group identifier to the policing database to retrieve first policing data and a second policeable group identifier;

producing a first policing result using the first policing data;

applying the second policeable group identifier to the policing database to retrieve second policing data; and

producing a second policing result using the second policing data.

7. The method of processing a packet of claim 5 wherein the policing results include one or more disposition recommendations, and the method further comprises the step of making a disposition decision for the packet using the policing results and at least one other disposition recommendation.

8. The method of processing a packet of claim 5 further comprising the step of combining the policing results into a single result by taking a worst case policing result.

9. A method for policing a data packet received by a data communication switch, the method comprising:

classifying the data packet into a plurality of

policeable groups;

identifying policing data associated with one or more policeable groups;

applying the policing data to produce one or more policing results for the policeable groups; and

recommending a disposition of the data packet from the policing results.

10. The method of claim 9 wherein a particular policeable group identifies a type of application to be policed.

11. The method of claim 9 wherein the policing data includes information on bandwidth constraints specified for at least one policeable group.

12. The method of claim 9 wherein the policing results indicate whether the data packet is to be forwarded.

13. The method of claim 9 wherein the policing results indicate whether the data packet is eligible to be dropped.

14. The method of claim 9 wherein the policing results indicate whether the data packet is to be dropped.

15. The method of claim 9 wherein the step of recommending a disposition comprises the step of combining the policing results to make a recommendation.

16. The method of claim 9 wherein the step of recommending a disposition comprises selecting one of the policing results as the recommended disposition.

17. The method of claim 9 further comprising the step of

updating the policing data based on the recommended disposition.

18. A method for policing a data packet received by a data communication switch, the method comprising the steps of:

creating a policing database including a plurality of policing data entries specifying policing data for a plurality of policeable groups;

applying a first identifier for retrieving a first policing data associated with a first policeable group and a second identifier identifying a second policeable group;

applying the first policing data to produce a first policing result;

applying the second identifier for retrieving a second policing data;

applying the second policing data to produce a second policing result; and

recommending a disposition of the data packet from the first and second policing results.

19. The method of claim 18 wherein a particular policeable group identifies a type of application to be policed.

20. The method of claim 18 wherein the policing data includes information on bandwidth constraints specified for the policeable group.

21. The method of claim 18 wherein the policing results indicate whether the data packet is to be forwarded.

22. The method of claim 18 wherein the policing results indicate whether the data packet is eligible to be dropped.

23. The method of claim 18 wherein the policing results

indicate whether the data packet is to be dropped.

24. The method of claim 18 wherein the step of recommending a disposition comprises the step of combining the first and second policing results to make a recommendation.

25. The method of claim 18 wherein the step of recommending a disposition further comprises selecting either the first or second policing result as the recommended disposition.

26. The method of claim 18 further comprising the step of updating the first or second policing data based on the recommended disposition.

27. A policing engine for a data communication node, wherein the policing engine classifies a packet into a plurality of policeable groups, and wherein the packet is compared for the respective ones of the policeable groups against respective ones of bandwidth contracts to produce respective ones of policing results.

28. A policing engine for a data communication node, wherein a first policeable group identifier is applied to a policing database to retrieve first policing data and a second policeable group identifier, wherein the first policing data is applied to produce a first policing result, and the second policeable group identifier is applied to the policing database to retrieve second policing data, wherein the second policing data is applied to produce a second policing result.

29. A packet processor comprising:
an input for receiving a packet;

policing means for classifying the packet into a plurality of policeable groups,

wherein the packet is compared against one or more bandwidth contracts defined for the policeable groups to produce one or more policing results.

30. The packet processor of claim 29 wherein the policing means include a policing database, a first policeable group identifier is applied to the policing database to retrieve first policing data and a second policeable group identifier, the first policing data is applied to produce a first policing result, the second policeable group identifier is applied to the policing database to retrieve second policing data, and the second policing data is applied to produce a second policing result.

31. The packet processor of claim 29 further comprising a disposition means for making a disposition decision for the packet, wherein the policing results include one or more disposition recommendations, and the disposition means use the policing results and at least one other disposition recommendation to make the disposition decision for the packet.

32. The packet processor of claim 29 wherein the policing results are combined into a single result by taking a worst case policing result.

33. The packet switching controller of claim 1, the packet switching controller further comprising a debiting element, wherein at least one bandwidth contract has an associated token bucket to represent available bandwidth under said bandwidth contract, and the debiting element determines, using the policing results, whether or not to debit the

associated token bucket.

34. The packet switching controller of claim 3, the packet switching controller further comprising a debiting element, wherein at least one bandwidth contract has an associated token bucket to represent available bandwidth under said bandwidth contract, and the debiting element defers debiting the associated token bucket with the packet size until the disposition engine provides the disposition decision to the debiting element to be used for determining whether or not to debit the associated token bucket.

35. The method of processing a packet of claim 5, wherein at least one bandwidth contract has an associated token bucket to represent available bandwidth under said bandwidth contract, and wherein the method further comprises determining, using the policing results, whether or not to debit the associated token bucket.

36. The method of processing a packet of claim 7, wherein at least one bandwidth contract has an associated token bucket to represent available bandwidth under said bandwidth contract, and wherein the method further comprises determining, using the disposition decision. whether or not to debit the associated token bucket with the packet size.

37. The method for policing a data packet of claim 11, further comprising the steps of:

generating a disposition decision for the data packet using the disposition recommendation from the policing results and at least one other disposition recommendation; and

determining whether or not to update the information on bandwidth constraints using the disposition decision.

38. The method for policing a data packet of claim 20, further comprising the steps of:

generating a disposition decision for the data packet using the disposition recommendation from the first and second policing results and at least one other disposition recommendation; and

determining whether or not to update the information on bandwidth constraints using the disposition decision.

39. The policing engine of claim 27, wherein whether or not bandwidth available under the bandwidth contracts are updated is determined based on the policing results.

40. The packet processor of claim 31, the packet processor further comprising debiting means, wherein at least one bandwidth contract has an associated token bucket to represent available bandwidth under said bandwidth contract, and the debiting means defers debiting the associated token bucket with the packet size until the disposition means provides the disposition decision to the debiting means to be used for determining whether or not to debit the associated token bucket.

41. A data policing method, the method comprising the steps of:

receiving a packet;

adding a time credit to a first token count to generate a second token count;

applying the second token count to generate a policing result for the packet;

applying the policing result to determine whether to subtract a size debit from the second token count to generate a third token count or not; and

subtracting the size debit from the second token count to generate a third token count if such subtraction has been determined through applying the policing result.

42. The data policing method of claim 41, the method further comprising the steps of:

receiving a second packet;

adding a second time credit to the second token count to generate a fourth token count if the third token count has not been generated;

adding a second time credit to the third token count to generate the fourth token count if the third token count has been generated; and

applying the fourth token count to generate a policing result for the second packet.

43. A data policing method, the method comprising the steps of:

receiving a packet;

adding a time credit to a first token count to generate a second token count;

applying the second token count to generate a policing result for the packet;

applying the policing result to generate a disposition result for the packet;

applying the disposition result to determine whether to subtract a size debit from the second token count to generate a third token count or not; and

subtracting the size debit from the second token count to generate the third token count if such subtraction has been determined through applying the disposition result.

44. The data policing method of claim 43, wherein the

policing result is applied as a recommendation with at least one other recommendation to generate the disposition result.

45. A data policing method, the method comprising the steps of:

receiving a packet;

adding a time credit to ones of token counts to generate respective ones of second token counts;

applying the ones of second token counts to generate a policing result for the packet;

applying the policing result to determine whether to subtract size debit from at least one of the second token counts to generate at least one third token count or not; and

subtracting the size debit from at least one of the second token counts to generate at least one third token count if such subtraction has been determined through applying the policing result.

46. A data policing method, the method comprising the steps of:

receiving a packet;

adding a time credit to ones of token counts to generate respective ones of second token counts;

applying the ones of second token counts to generate a policing result for the packet;

applying the policing result to generate a disposition result for the packet;

applying the disposition result to determine whether to subtract or not a size debit from at least one of the second token counts to generate at least one third token count; and

subtracting the size debit from at least one of the second token counts to generate at least one third token count if such subtraction has been determined through applying the disposition result.

3. Detailed Description of Invention

CROSS-REFERENCE TO RELATED APPLICATION(S)

The present application claims the priority of U.S. Provisional Application No. 60/206,617 entitled "System and Method for Enhanced Line Cards" filed May 24, 2000, U.S. Provisional Application No. 60/206,996 entitled "Flow Resolution Logic System and Method" filed May 24, 2000 and U.S. Provisional Application No. 60/220,335 entitled "Programmable Packet Processor" filed July 24, 2000, the contents of all of which are fully incorporated by reference herein. The present application contains subject matter related to the subject matter disclosed in U.S. Patent Application No. 09/751,194 entitled "Programmable Packet Processor with Flow Resolution Logic" filed December 28, 2000, the contents of which are fully incorporated by reference herein.

FIELD OF THE INVENTION

This invention relates generally to data communication switches, and more particularly to a data communication switch employing multiple levels of rate policing on a data packet.

BACKGROUND OF THE INVENTION

Rate policing is increasingly becoming important in data communication networks as customers entitled to different qualities of service (QoS) compete for the available bandwidth of a common set of network resources. Rate policing is typically accomplished at each switch by classifying each packet into a single policy group and comparing the classified packet against one or more bandwidth contracts defined for the group. Based on the identified bandwidth contract, the packet may be forwarded, be forwarded with a discard eligible marking, or be discarded.

Existing rate policing methods typically police data traffic on a per-port basis with no regard to other information about the traffic. Data exceeding the rate subscribed by the customer is typically marked to be dropped if congestion occurs. Thus, a customer typically has no flexibility to selectively drop certain data based on the data type, such as based on the particular application associated with the data.

With the increasing desire to tailor communication networks to the individualized needs of customers, it is desirable to provide policing logic that has increased flexibility, but whose implementation is not so complex as to substantially reduce line speed.

SUMMARY OF THE INVENTION

In one embodiment of the present invention, a packet switching controller is provided. The packet switching controller includes an input for receiving a packet and a policing element for classifying the packet into a plurality of policeable groups. The packet is compared against one or more bandwidth contracts defined for the policeable groups to produce one or more policing results.

In another embodiment of the present invention, a method of processing a packet is provided. A packet is received and classified into a plurality of policeable groups. The packet is compared against one or more bandwidth contracts defined for the policeable groups to produce one or more policing results.

In yet another embodiment of the present invention, a method for policing a data packet received by a data communication switch is provided. The data packet is classified into a plurality of policeable groups. Then, policing data associated with one or more policeable groups is identified. The policing data is applied to produce one or more policing results for the policeable groups, and a disposition of the data

packet is recommended from the policing results.

In still another embodiment of the present invention, a method for policing a data packet received by the data communication switch is provided. A policing database including a plurality of policing data entries specifying policing data for a plurality of policeable groups is created. A first identifier is applied for retrieving a first policing data associated with a first policeable group and a second identifier identifying a second policeable group. Then, the first policing data is applied to produce a first policing result. Further, the second identifier is applied for retrieving a second policing data. Then, the second policing data is applied to produce a second policing result. A disposition of the data packet is recommended from the first and second policing results.

In a further embodiment of the present invention, a policing engine for a data communication node is provided. The policing engine classifies a packet into a plurality of policeable groups. The packet is compared for the respective ones of the policeable groups against respective ones of bandwidth contracts to produce respective ones of policing results.

In a still further embodiment of the present invention, a policing engine for a data communication node is provided. A first policeable group identifier is applied to a policing database to retrieve first policing data and a second policeable group identifier. The first policing data is applied to produce a first policing result, and the second policeable group identifier is applied to the policing database to retrieve second policing data. The second policing data is applied to produce a second policing result.

In a yet further embodiment of the present invention, a packet processor is provided. The packet processor includes an input for receiving a packet and policing means for classifying the packet into a plurality of policeable groups. The packet is compared against one or more bandwidth contracts defined for the policeable groups to produce one or more policing results.

I. Overview

In FIG. 1, network environment including a packet switching node 10 is illustrated. The packet switching node may also be referred to as a switch, a data communication node or a data communication switch. The packet switching node 10 includes switching interfaces 14, 16 and 18 interconnected to respective groups of LANs 30, 32, 34, and interconnected to one another over data paths 20, 22, 24 via switching backplane 12. The switching backplane 12 preferably includes switching fabric. The switching interfaces may also be coupled to one another over control paths 26 and 28.

The switching interfaces 14, 16, 18 preferably forward packets to and from their respective groups of LANs 30, 32, 34 in accordance with one or more operative communication protocols, such as, for example, media access control (MAC) bridging and Internet Protocol (IP) routing. The switching node 10 is shown for illustrative purposes only. In practice, packet switching nodes may include more or less than three switching interfaces.

FIG. 2 is a block diagram of a switching interface 50 in one embodiment of the present invention. The switching interface 50 may be similar, for example, to the switching interfaces 14, 16, 18 of FIG. 1. The switching interface 50 includes an access controller 54 coupled between LANs and a packet switching controller 52. The access controller 54, which may, for example, include a media access controller (MAC), preferably receives inbound packets off LANs, performs flow-independent physical and MAC layer operations on the inbound packets and transmits the inbound packets to the packet switching controller 52 for flow-dependent processing. The access controller 54 preferably also receives outbound packets from the packet switching controller 52 and transmits the packets on LANs. The access controller 54 may also perform physical and MAC layer operations on the outbound packets prior

to transmitting them on LANs.

The packet switching controller 52 preferably is programmable for handling packets having wide variety of communications protocols. The packet switching controller 52 preferably receives inbound packets, classifies the packets, modifies the packets in accordance with flow information and transmits the modified packets on switching backplane, such as the switching backplane 12 of FIG. 1. The packet switching controller 52 preferably also receives packets modified by other packet switching controllers via the switching backplane and transmits them to the access controller 54 for forwarding on LANs. The packet switching controller 52 may also subject selected ones of the packets to egress processing prior to transmitting them to the access controller 54 for forwarding on LANs.

FIG. 3 is a block diagram of a programmable packet switching controller 100 in one embodiment of the present invention. The programmable packet switching controller 100, for example, may be similar to the packet switching controller 52 of FIG. 2. The programmable packet switching controller 100 preferably has flow resolution logic for classifying and routing incoming flows of packets. Due to its programmable nature, the programmable packet switching controller preferably provides flexibility in handling many different protocols and/or field upgradeability. The programmable packet switching controller may also be referred to as a packet switching controller, a switching controller, a programmable packet processor, a network processor, a communications processor or as another designation commonly used by those skilled in the art.

The programmable packet switching controller 100 includes a packet buffer 102, a packet classification engine 104, an application engine 106 and a policing engine 120. The policing engine may also be referred to as a policing element. Packet

switching controllers in other embodiments may include more or less components. For example, a packet switching controller in another embodiment may include a pattern match module for comparing packet portions against a predetermined pattern to look for a match. The packet switching controller in yet another embodiment may include an edit module for editing inbound packets to generate outbound packets.

The programmable packet switching controller 100 preferably receives inbound packets 108. The packets may include, but are not limited to, Ethernet frames, ATM cells, TCP/IP and/or UDP/IP packets, and may also include other Layer 2 (Data Link/MAC Layer), Layer 3 (Network Layer) or Layer 4 (Transport Layer) data units. For example, the packet buffer 102 may receive inbound packets from one or more Media Access Control (MAC) Layer interfaces over the Ethernet.

The received packets preferably are stored in the packet buffer 102. The packet buffer 102 may include a packet FIFO for receiving and temporarily storing the packets. The packet buffer 102 preferably provides the stored packets or portions thereof to the packet classification engine 104 and the application engine 106 for processing.

The packet buffer 102 may also include an edit module for editing the packets prior to forwarding them out of the switching controller as outbound packets 118. The edit module may include an edit program construction engine for creating edit programs real-time and/or an edit engine for modifying the packets. The application engine 106 preferably provides application data 116, which may include a disposition decision for the packet, to the packet buffer 102, and the edit program construction engine preferably uses the application data to create the edit programs. The outbound packets 118 may be transmitted over a switching fabric interface to communication networks, such as, for example, the Ethernet.

The packet buffer 102 may also include either or both a header data extractor and a header data cache. The header data extractor preferably is used to extract one or more fields from the packets, and to store the extracted fields in the header data cache as extracted header data. The extracted header data may include, but are not limited to, some or all of the packet header. In an Ethernet system, for example, the header data cache may also store first N bytes of each frame.

The extracted header data preferably is provided in an output signal 110 to the packet classification engine 104 for processing. The application engine may also request and receive the extracted header data over an interface 114. The extracted header data may include, but are not limited to, one or more of Layer 2 MAC addresses, 802.1P/Q tag status, Layer 2 encapsulation type, Layer 3 protocol type, Layer 3 addresses, ToS (type of service) values and Layer 4 port numbers. In other embodiments, the output signal 110 may include the whole inbound packet, instead of or in addition to the extracted header data. In still other embodiments, the packet classification engine 104 may be used to edit the extracted header data to be placed in a format suitable for use by the application engine, and/or to load data into the header data cache.

The packet classification engine 104 preferably includes a programmable microcode-driven embeddded processing engine. The packet classification engine 104 preferably is coupled to an instruction RAM (IRAM) (not shown). The packet classification engine preferably reads and executes instructions stored in the IRAM. In one embodiment, many of the instructions executed by the packet classification engine are conditional jumps. In this embodiment, the classification logic includes a decision tree with leaves at the end points that preferably indicate different types of packet classifications. Further, branches of the decision tree preferably are selected based on comparisons

between the conditions of the instructions and the header fields stored in the header data cache. In other embodiments, the classification logic may not be based on a decision tree.

In one embodiment of the present invention, the application engine 106 preferably has a pipelined architecture wherein multiple programmable sub-engines are pipelined in series. Each programmable sub-engine preferably performs an action on the packet, and preferably forwards the packet to the next programmable sub-engine in a "bucket brigade" fashion. The packet classification engine preferably starts the pipelined packet processing by starting the first programmable sub-engine in the application engine using a start signal 112. The start signal 112 may include identification of one or more programs to be executed in the application engine 106. The start signal 112 may also include packet classification information. The programmable sub-engines in the application engine preferably have direct access to the header data and the extracted fields stored in the header data cache over the interface 114.

The application engine may include other processing stages not performed by the programmable sub-engines, however, the decision-making stages preferably are performed by the programmable sub-engines to increase flexibility. In other embodiments, the application engine may include other processing architectures.

The disposition decision included in the application data 116 preferably is also provided to the policing engine 120. The policing engine 120 preferably also receives one or more policing IDs 124. The policing engine 120 preferably uses the disposition decision and the policing IDs to generate one or more policing recommendations 122. The policing recommendations may be a type of disposition recommendation, and may also be referred to as policing results. The policing recommendations preferably are provided to the application engine 106 to be used together with

other disposition recommendations to generate application data, which may include the disposition decision.

II. Programmable Disposition Logic

FIG. 4 is a block diagram of a packet switching controller 130 with programmable disposition logic. The packet switching controller 130 may be similar, for example, to the packet switching controller 100 of FIG. 3. The packet switching controller includes a packet buffer 132, a packet classification engine 134, a pattern match lookup logic 136, an application engine 138 and a policing engine 166.

The application engine includes a source lookup engine 140, a destination lookup engine 142 and a disposition engine 144. The packet classification engine, the source lookup engine, the destination lookup engine and the disposition engine preferably are programmable with one or more application programs. In other words, each of the packet classification engine and the sub-engines of the application engine preferably includes a programmable microcode-driven embedded processing engine. In other embodiments, one or more of these engines may be implemented in hardware, i.e., as hardwired logic. The policing engine 166 may be implemented in hardwired logic or in programmable microcode-driven embedded processing engine.

The packet buffer 132 preferably receives and stores inbound packets 146. The packet buffer preferably provides the inbound packets or portions thereof 148 to the packet classification engine 134. The packet classification engine preferably classifies the packets using its application programs programmed thereon, and preferably provides a program identification 152 to the application engine 138. More particularly, the program identification 152 preferably is provided to the source lookup engine 140, the destination lookup engine 142 and the disposition engine 144 in the application

engine. In one embodiment of the present invention, the packet classification engine 134 includes a decision tree-based classification logic.

The program identification 152 preferably is used to select application programs to be executed in each of the source lookup engine, the destination lookup engine and the disposition engine. The application programs to be executed in the source lookup engine, the destination lookup engine and the disposition engine preferably are selected based at least partly on packet classification information. The packet classification information may also be provided together with the program identification.

The packet buffer preferably also provides the inbound packets or portions thereof 150 to the pattern match lookup logic 136. The pattern match lookup logic preferably includes a predefined pattern against which the packets or the packet portions are compared. For example, the packet portions used for pattern matching may include portions of packet header data, packet payload data, or both the packet header data and the packet payload data. In other embodiments, the predefined pattern may reside in an external memory, which is accessed by the pattern match lookup logic for pattern matching. In still other embodiments, the match pattern may change during the operation of the packet switching controller.

After a comparison is made, a result 154 of the comparison preferably is provided to the application engine 138. More particularly, the result 154 of the comparison preferably is provided to the disposition engine 144 in the application engine. In some embodiments, the result may be provided to the disposition engine only when there is a match.

The source lookup engine 140 preferably generates a disposition recommendation 160 for an inbound packet at least partly by performing a source address lookup using a source

address of the inbound packet. The disposition recommendation 160 preferably also depends on the application program executed in the source lookup engine 140 in accordance with the program identification provided by the packet classification engine. The disposition recommendation 160 preferably includes a security recommendation for the inbound packet.

In other embodiments, the source lookup engine 140 may be used to build one or more keys, which may then be used to look up the source address (e.g., IPSA) of the inbound packet in an address table. The keys may include, but are not limited to, one or more of Virtual LAN Identification (VLAN ID), application identification (APP ID) and IPSA. One or more keys built by the source lookup engine 140 may also be used to formulate a disposition recommendation, such as, for example, the security recommendation.

The destination lookup engine 142 preferably receives an output 156 from the source lookup engine 140. The output 156 may include the key used to look up the source address and/or the result of the lookup. The destination lookup engine preferably executes its application program identified by the packet classification engine 134 and generates one or more police identifiers (IDs) 168. The police IDs 168 may be based at least partly on destination address lookup using a destination address of the inbound packet.

The policing engine 166 preferably uses the police IDs 168 as keys to access policing data in a policing data table. The policing engine 166 preferably uses the accessed policing data to generate one or more policing recommendations 170. The policing recommendations preferably are used by the disposition engine along with other disposition recommendations to generate application data, which may include the disposition decision. When the pattern match lookup logic 136 finds a match, the pattern match result 154 preferably overrides the policing

recommendations. The policing recommendations preferably are used to generate a single recommendation by selecting the worst case policing recommendation. The policing engine may also perform accounting functions.

In other embodiments, the destination lookup engine 142 may be used to build one or more keys, which may then be used to look up the destination address (e.g., IPDA) of the inbound packet in an address table. The keys may include, but are not limited to, one or more of Virtual LAN Identification (VLAN ID), application identification (APP ID) and IPDA.

The disposition engine 144 preferably receives a number of disposition recommendations including, but not limited to, the security recommendation in the disposition recommendation 160, the policing recommendation 170, and the pattern match result 154. The disposition engine preferably generates a disposition decision 162 based on the disposition recommendations as well as the packet classification and/or program identification. The disposition decision 162 may include one of the disposition recommendations. In general, the pattern match result 154 may override the policing recommendation 170, and the policing recommendation may override the security recommendation in the disposition recommendation 160. The disposition decision 162 may be a part of application data, which may include, but is not limited to, one or more of accounting data, routing data and policing data.

The disposition decision preferably is provided to the packet buffer to be used for editing the inbound packets to be provided as outbound packets 164. The disposition decision preferably is also fed back to the policing engine for policing and accounting. For example, when the inbound packet is dropped, the policing engine should be made aware of that fact. In other embodiments, the destination lookup engine may include the policing engine. In such cases, the disposition decision

preferably is provided to the destination lookup engine for policing and accounting.

FIG. 5 is a flow diagram of a process of programmatically generating a disposition decision using multiple disposition recommendations and classification information. In step 180, a packet buffer, such as, for example, the packet buffer 132 of FIG. 4, preferably receives an inbound packet. In the packet buffer, packet header data may be extracted and stored in a header data cache.

The inbound packet or a portion of the inbound packet, which may include the header data, preferably is provided to a pattern match lookup logic, such as, for example, the pattern match lookup logic 136 of FIG. 4. In step 182, the pattern match lookup logic preferably performs a pattern match lookup between the inbound packet or the portion of the inbound packet and a predetermined pattern to generate a pattern match recommendation as indicated in step 188. The predetermined pattern, for example, may be contained in an internal or external memory. In other embodiments, the match pattern may change dynamically.

Meanwhile, the inbound packet or a portion thereof preferably is also provided to a packet classification engine, such as, for example, the packet classification engine 134 of FIG. 4. In step 184, the packet classification engine preferably classifies the packet and preferably identifies application programs based on the packet classification. In step 186, the program identification preferably is provided to a source lookup engine, a destination lookup engine and a disposition engine in an application engine, such as, for example, the application engine 138 of FIG. 4. The program identification preferably indicates application programs to be executed in these sub-engines. The packet classification information preferably is also provided to the source lookup

engine, the destination lookup engine and the disposition engine. The source lookup engine preferably generates a security recommendation in step 190, while the policing engine preferably generates a policing recommendation in step 192 using police IDs from the destination lookup engine.

In step 194, the pattern match recommendation, the security recommendation and the policing recommendation preferably are provided to the disposition engine. The disposition engine preferably generates a disposition decision using one or more of the selected application program and the disposition recommendations. The disposition decision preferably is provided to the packet buffer to be used for editing and transmission of the inbound packet as an outbound packet in step 196. In step 198, the disposition decision preferably is also fed back to the policing engine for operations such as, for example, policing and accounting.

III. Multi-Level Policing

In one embodiment of the present invention, the policing engine preferably employs multi-level policing logic for policing the traffic flowing through the packet switching controller based on multiple policy groups. A customer preferably specifies the applicable policy groups and bandwidths applicable to those groups in her bandwidth contract. In an exemplary scenario, the customer may specify in her bandwidth contract that she will pay for 1 Gbps of data traffic on a particular port. The customer may further assign different data flow limits to the subnets in her company. For example, the customer may limit the engineering subnet to 300 Mbps and the accounting subnet to 100 Mbps. Furthermore, the customer may specify that web traffic is to be limited to 200 Mbps for the entire company. Thus, instead of policing the traffic solely on a per-port basis with no regard to the type of traffic, web

traffic and traffic originating from the engineering or accounting subnets may be identified and policed based on their respective thresholds.

Further, a bandwidth contract between service provider and customer may also determine QoS actions. The QoS actions preferably identify QoS applicable to the traffic meeting the flow conditions. The QoS actions may indicate a maximum bandwidth, minimum bandwidth, peak bandwidth, priority, latency, jitter, maximum queue depth, maximum queue buffers, and the like.

The bandwidth policing function preferably controls the ingress data rate on a per-flow bases as part of a general solution to limit, e.g., police, and shape traffic flows. FIG. 6 is a block diagram illustrating policing of different flows. The policing parameters preferably are established by defining a Committed Information Rate (CIR) in units of bytes per time along with a Committed Burst Size (CBS) and Excess Burst Size (EBS) both in units of bytes. The packets preferably are classified, i.e., marked, into a first bucket (Drop Eligible (DE) bucket) 200 and a second bucket (Drop bucket) 202.

As packets are presented at a given ingress rate, they preferably are marked according to a current balance within each bucket and its relationship to the CBS and EBS. The first bucket preferably maintains a Discard Eligible (DE) balance. The second bucket preferably maintains a Drop balance. If the ingress rate is less than the CBS, the packets preferably are marked as Forward. If the ingress rate is greater than or equal to the CBS but below the EBS, packets preferably are marked as DE. If the ingress rate is greater than or equal to the EBS, packets preferably are marked as Drop.

FIG. 7 is a policing data table 250 used for policing data packets based on multiple policy levels in one embodiment of the present invention. The policing data table 250 may be stored

in a policing engine, which may be similar to the policing engine 166 of FIG. 4. The policing data table 250 may also be referred to as a policing database.

The policing data table 250 includes policing data for performing checks of the current rate of traffic flowing through a packet switching controller, such as, for example, the packet switching controller 130 of FIG. 4. The policing data table 250 may be arranged in a variety of ways, but preferably is configured as sequential entries, with each entry providing policing data 252 that is associated with a particular policy group. Each policing data 252 preferably is identified by a unique police identifier (ID)/key 254.

The police ID 254 preferably identifies different policy groups to which the packet may be classified. Preferably, each police ID 254 is composed of a customer identifier 254a and/or an application identifier 254b. The customer identifier preferably identifies a particular customer based on source address, physical port, or the like. The application identifier 254b preferably is an internal identifier assigned by an application RAM based on the type of application associated with the packet. Exemplary applications include web applications, Voice over IP (VoIP) applications, and the like.

A next police ID 256 preferably allows nested lookups in the policing database to identify additional policy groups applicable to the packet. The policing data 252 associated with those policy groups preferably are also retrieved for performing a rate check for the current packet.

Each policing data 252 preferably depicts the current bandwidth as well as the bandwidth limits of each policy group identified by the police ID 254. A Drop balance 252c and a Drop Eligible (DE) balance 252d preferably maintain counts of the amount of traffic flowing through the packet switching controller. The Drop and DE balances 252c, 252d preferably are

respectively compared against a Drop and DE limits 252e, 252f for recommending that the current packet be forwarded, forwarded with a DE marking, or dropped immediately. The Drop balance 252c preferably is not incremented until the DE balance 252d is greater than a DE limit 252f.

Each policing data 252 preferably further includes a timestamp 252b indicative of a time at which a last balance calculation was done. Given a current time and the timestamp information, an elapsed time from the last balance calculation may be measured for calculating a rate of traffic during this time. The size of the timestamp increments may be adjusted based on a budget (CIR) 252a value also maintained in the policing data table 250. For example, the budget value may be defined as bytes per timestamp increment in one embodiment of the present invention.

In the illustrated policing data table 250, the policing engine preferably performs a rate check 258 or 260 based on a first police ID to produce a first policy result indicating the recommended disposition of the packet. The policing engine preferably further determines if the packet is to be policed based on additional policy groups. In doing so, the policy engine preferably examines the next police ID field 256 and retrieves the policing data identified by the ID. A second rate check 262 preferably is then performed on the same packet to produce a second policy result based on the second rate check. Additional rate checks may continue to be performed based on values on the next policy ID field 256. In one embodiment of the present invention, up to four policing algorithms may be executed for each packet while maintaining line rate performance. In other embodiments, more or less than four policing algorithms may be executed.

FIG. 8 is an exemplary flow diagram of a multi-level policing process. The process starts, and in step 300, the

policing engine preferably receives a new police ID for an incoming packet. In step 302 the policing engine preferably retrieves the policing data associated with the police ID. In step 304, the policing engine preferably calculates a new Drop or DE balance, preferably according to the following formula:

$$Balance_{new} = Balance_{old} - [budget * (time - timestamp)] + packetsize$$

In the formula, $Balance_{new}$ and $Balance_{old}$ preferably represent new and current balances, respectively, for either the Drop bucket or DE bucket associated with the police ID. Budget preferably represents budget 252a, e.g., CIR, associated with the police ID. The current Drop and DE balances correspond to DROP BAL 252c and DE BAL 252d, respectively. Time and timestamp, respectively, preferably represent current time and timestamp 252b associated with the police ID. Packetsize preferably represents size of the packet being processed.

In step 306, the new Drop balance or DE balance is applied towards the Drop limit 252e or DE limit 252f. The balance preferably is applied towards the DE balance until the DE limit has been exceeded. The policing engine preferably compares the DE balance against the DE limit and preferably determines that the packet is to be forwarded if the DE balance is less than the DE limit. If the DE balance exceeds the DE limit, the balance preferably is applied towards the Drop balance. The policing engine preferably then compares the Drop balance against the Drop limit, and preferably determines that the packet is to be forwarded with a DE marking if the Drop balance is less than the Drop limit. However, if the Drop limit has been exceeded, the policing engine preferably determines that the packet is to be discarded immediately.

For example, in practice, the new balances preferably are calculated and then compared against the DE and Drop limits to

determine forwarding status. The balances preferably are updated based on the forwarding result. For example, if the packet is marked Forward, the DE balance preferably is updated. In other words, when the packet is marked Forward, the DE bucket, such as, for example, the first bucket 200 of FIG. 6, preferably is filled. For further example, if the packet is marked DE, the Drop balance preferably is updated. In other words, when the packet is marked DE, the Drop bucket, such as, for example, the second bucket 202 of FIG. 6, is filled. At this point, the DE bucket is already full. For still further example, if the packet is marked Drop, neither the DE balance nor the Drop balance is updated since both buckets are full at this point.

In step 308, a determination is made as to whether there are additional police IDs indicated for the current packet. If there are, the process returns to step 302 to retrieve the policing data identified by the additional police IDs and to produce additional policy results.

In step 310, the policing engine preferably notifies a disposition engine, such as, for example, the disposition engine 144 of FIG. 4, of the policing results, which may also be referred to as policing recommendations. In the event that multiple policy results are available for the packet being processed, the policing engine preferably selects the most conservative policing result, i.e., worst case policing result, and preferably returns that result to the disposition engine. The disposition engine preferably uses the police results and other disposition recommendations, e.g., security recommendation and pattern match result, to generate a disposition decision.

In step 312, the policing engine preferably receives notice from the disposition engine of the disposition decision. The disposition decision may include the decision on whether the packet was forwarded, forwarded with a DE marking, or dropped.

In step 314, the policing engine preferably determines whether the packet was forwarded. If it was, each policing data associated with the forwarded packet is updated in step 316 to reflect an increased traffic.

The values updated in the police database preferably include one or more of the DE balance, the Drop balance and the timestamp. The DE balance preferably is updated if it is less than the DE limit. The Drop balance preferably is updated if the DE balance is greater than the DE limit and the Drop balance is less than the Drop limit. If both balances are over their respective limits, then preferably neither is updated. In any case, it is desirable to not add the 'packetsize' (size of the packet) value to either balance if the packet, e.g., frame, is dropped for any reason as indicated by the disposition decision, for example. This way, an accurate count preferably is made of the packets coming into the switching fabric.

IV. Flow Rate Policing with Deferred Debiting

In one embodiment of the present invention, deferred debiting preferably is used with flow rate policing. FIG. 9 is a block diagram 400 of a packet switching controller having flow rate policing with deferred debiting in this embodiment of the present invention. The deferred debiting may be used in conjunction with the multi-level policing logic.

As shown in FIG. 9, a field extractor 402 receives packets, provides flow information to generic decision logic 408 and deferred debit policing logic 410, and provides the packet to a packet size calculator 404. The packet size calculator 404 provides output to a packet size buffer 406 and provides the packet to a packet buffer 412. The generic decision logic 408 and the deferred debit policing logic 410, respectively, provide a generic decision result and a policing result to disposition logic 414, which provides a disposition result to the packet

buffer 412. The disposition logic 414 also provides the disposition result to the deferred debit policing logic 410, which uses the disposition result and the packet size information for deferred debiting.

Flow rate policing has become increasingly important in data communication networking as customers entitled to different qualities of service compete for shared network bandwidth. Flow rate policing typically involves comparing packets within a flow against one or more bandwidth contracts defined for the flow to resolve whether to: (i) admit the packet without conditions; (ii) admit the packet with conditions (e.g. mark the packet discard eligible); or (iii) discard the packet.

Flow rate policing schemes typically maintain a "token bucket" to express the currently available bandwidth under each bandwidth contract. Typically, a packet is deemed to be within a flow's bandwidth contract if there are presently enough tokens in the bucket maintained for the contract; a packet is deemed to exceed the contract if there are not presently enough tokens in the bucket maintained for the contract. Tokens are added to the bucket as time elapses via time credits; tokens are subtracted from the bucket as packets are admitted via packet size debits.

A common expression used to maintain token bucket state is:

$TC_{new} = TC_{old} + C - D$

where

TC_{new} = new token count

TC_{old} = old token count

C = time credit

D = size debit

A single instance of the token bucket state expression may be applied to effectuate simple admit/discard policing decisions

as follows. When a packet within a flow arrives for a policing decision, a new token count TC_{new} for the flow's bandwidth contract is calculated by adding a time credit C reflecting the elapsed time since the policing decision on the previous packet and by subtracting a size debit D reflecting the size of the current packet. The new token count TC_{new} for the flow's bandwidth contract is then compared with zero. If the new token count TC_{new} is greater than or equal to zero, the current packet is within the bandwidth contract and is admitted. If the new token count TC_{new} is less than zero, the current packet exceeds the bandwidth contract and is discarded.

Two instances of the token bucket state expression may be applied to the same flow to provide more sophisticated policing decisions. For instance, a discard token bucket and a discard eligible token bucket may be separately maintained for a flow. In that event, if the new discard token count TC_{new-de} is greater than or equal to zero but the new discard token count TC_{new-d} is less than zero, the current packet is within the discard bandwidth contract but exceeds the discard eligible bandwidth contract. Accordingly, the current packet is admitted (since it is within the drop bandwidth contract) subject to the condition that it be marked as discard eligible (since it exceeds the discard eligible bandwidth contract). Such a three-level "dual token bucket" policing scheme is described in IETF Request for Comment 2697 entitled "A Single Rate Three Color Marker".

Applying the token bucket state expression to police high speed data flows in state of the art packet switching controllers has met with some practical difficulty, particularly with regard to the teaching to subtract the size debit D reflecting the size of the current packet prior to making the policing decision. First, the current packet's size may be determined external to the policing logic. Thus, the size debit D for the current packet may not be available at the time the

policing decision is made. Second, the policing decision alone may not dictate the final disposition of the packet. Thus, deduction of the size debit D for the current packet may require later reversal. Third, the size debit D for the current packet, if deducted prior to making the policing decision, will result in the current packet being found to exceed a bandwidth contract even though there are enough tokens in the bucket to accommodate most (but not all) of the packet.

On the other hand, the practical benefit of deducting the size debit D for the current packet prior to making the policing decision is not clear, since in high speed controllers the data transfer rate is exponentially larger than the maximum packet size. At most a nominal and temporary violation of the bandwidth contract for a flow will occur as long as the size debit D is made within a reasonable time thereafter.

In this embodiment of the present invention, deferred debiting preferably is used to overcome the above difficulties in applying the common token bucket state expression to police high speed data flows.

For example, a data policing method may be provided. The data policing method preferably includes: receiving a packet; adding a time credit to a first token count to generate a second token count; applying the second token count to generate a policing result for the packet; and applying the policing result for the packet to subtract or not a size debit from the second token count to generate or not, respectively, a third token count.

The data policing method may further comprise: receiving a second packet; adding a time credit to the second token count to generate a fourth token count; and applying the fourth token count to generate a policing result for the second packet.

Another data policing method may also be provided. This data policing method preferably includes: receiving a packet;

adding a time credit to a first token count to generate a second token count; applying the second token count to generate a policing result for the packet; applying the policing result for the packet to generate a disposition result for the packet; and applying the disposition result for the packet to subtract or not a size debit from the second token count to generate or not, respectively, a third token count.

In this data policing method, the police result may be applied as a recommendation with at least one other recommendation to generate the disposition result for the packet.

Yet another data policing method preferably includes: receiving a packet; adding a time credit to ones of token counts to generate respective ones of second token counts; applying the ones of second token counts to generate a policing result for the packet; and applying the policing result for the packet to subtract or not a size debit from at least one of the second token counts to generate or not, respectively, at least one third token count.

Still another data policing method preferably includes: receiving a packet; adding a time credit to ones of token counts to generate respective ones of second token counts; applying the ones of second token counts to generate a policing result for the packet; applying the policing result for the packet to generate a disposition result for the packet; and applying the disposition result for the packet to subtract or not a size debit from at least one of the second token counts to generate or not, respectively, at least one third token count.

The following data policing methods further illustrate flow rate policing with deferred debiting in one embodiment of the present invention.

A data policing method preferably includes: receiving a packet; adding a time credit to a first token count to generate

a second token count; applying the second token count to generate a policing result for the packet; and applying the policing result to subtract or not a size debit from the second token count to generate or not, respectively, a third token count.

The data policing method preferably further includes: receiving a second packet; adding a time credit to the second token count to generate a fourth token count; and applying the fourth token count to generate a policing result for the second packet.

Another data policing method preferably includes: receiving a packet; adding a time credit to a first token count to generate a second token count; applying the second token count to generate a policing result for the packet; applying the policing result to generate a disposition result for the packet; and applying the disposition result to subtract or not a size debit from the second token count to generate or not, respectively, a third token count. The police result may be applied as a recommendation with at least one other recommendation to generate the disposition result.

Yet another data policing method preferably includes: receiving a packet; adding a time credit to ones of token counts to generate respective ones of second token counts; applying the ones of second token counts to generate a policing result for the packet; and applying the policing result to subtract or not a size debit from at least one of the second token counts to generate or not, respectively, at least one third token count.

Still another data policing method preferably includes: receiving a packet; adding a time credit to ones of token counts to generate respective ones of second token counts; applying the ones of second token counts to generate a policing result for the packet; applying the policing result to generate a disposition result for the packet; and applying the disposition

result to subtract or not a size debit from at least one of the second token counts to generate or not, respectively, at least one third token count.

Although this invention has been described in certain specific embodiments, those skilled in the art will have no difficulty devising variations which in no way depart from the scope and spirit of the present invention. It is therefore to be understood that this invention may be practiced otherwise than is specifically described. Thus, the present embodiments of the invention should be considered in all respects as illustrative and not restrictive, the scope of the invention to be indicated by the appended claims and their equivalents rather than the foregoing description.

4. Brief Description of Drawings

FIG. 1 illustrates a network environment including a packet switching node in which one embodiment of the present invention is used.

FIG. 2 is a block diagram of a switching interface in one embodiment of the present invention.

FIG. 3 is a block diagram of a programmable packet switching controller in one embodiment of the present invention;

FIG. 4 is a block diagram of a packet switching controller with programmable disposition logic in one embodiment of the present invention.

FIG. 5 is a flow diagram of a process of programmatically generating a disposition decision using multiple disposition recommendations and classification information in one embodiment of the present invention.

FIG. 6 is a block diagram illustrating the process of marking packets into different classifications.

FIG. 7 is a policing data table used for policing data packets based on multiple policy levels in one embodiment of the present invention.

FIG. 8 is a flow diagram of multi-level policing process in one embodiment of the present invention.

FIG. 9 is a block diagram of a packet switching controller having flow rate policing with deferred debiting in one embodiment of the present invention.

Fig. 1

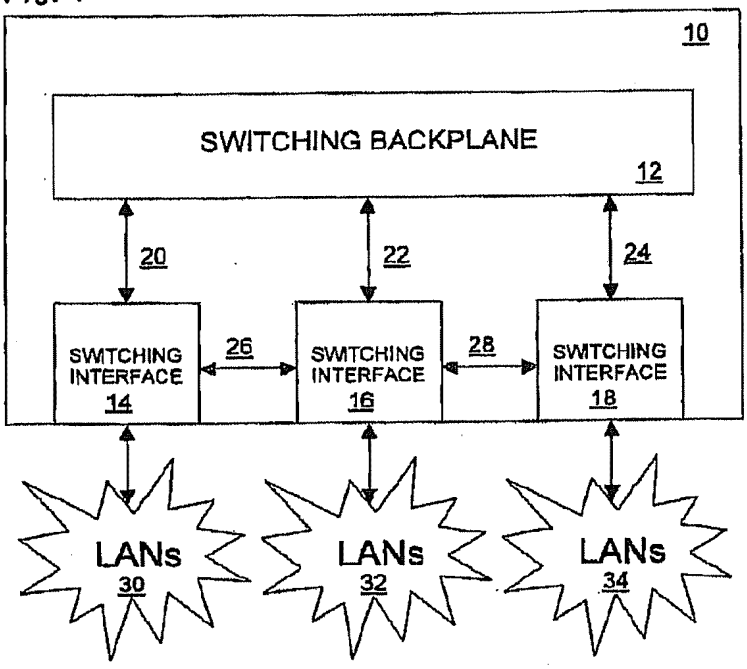

FIG. 1

Fig. 2

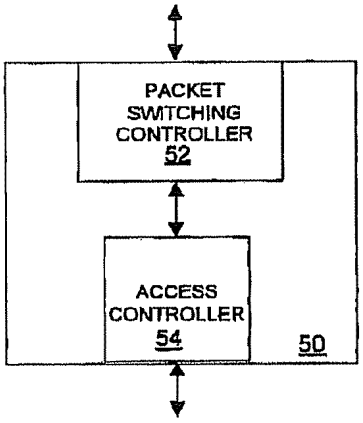

FIG. 2

Fig. 3

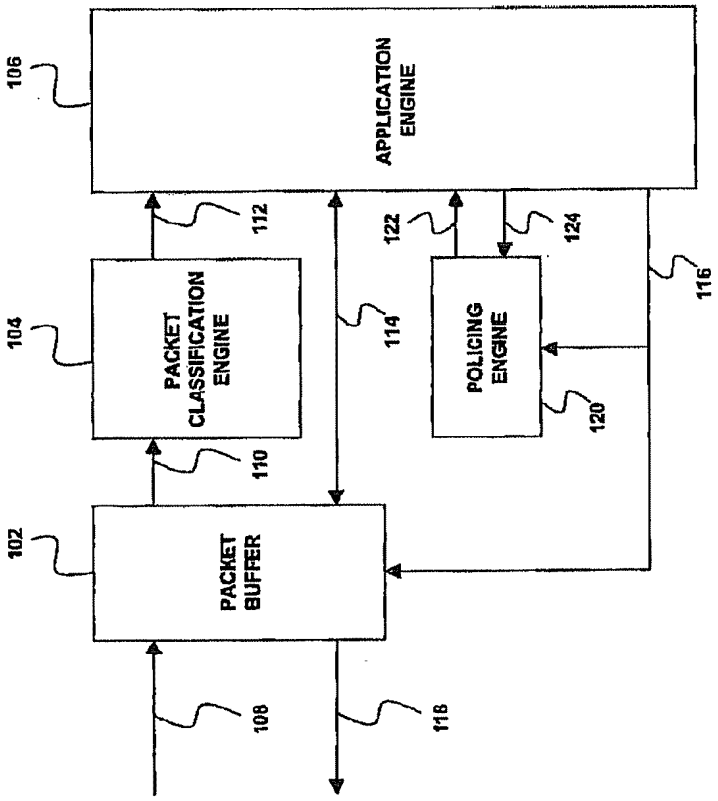
APPLICATION ENGINE
106
PACKET CLASSIFICATION ENGINE
104
POLICING ENGINE
PACKET BUFFER
102
108
110
112
114
116
118
120
122
124
100

FIG. 3

Fig. 4

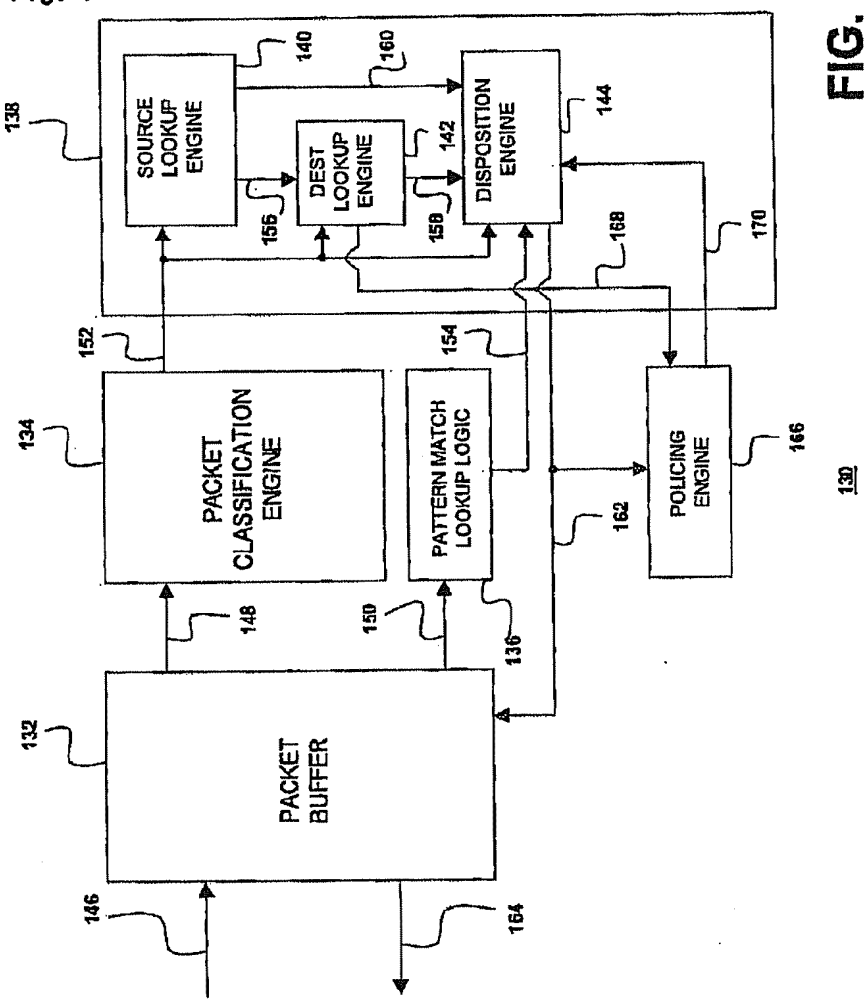
SOURCE LOOKUP ENGINE
DEST LOOKUP ENGINE
DISPOSITION ENGINE
PACKET CLASSIFICATION ENGINE
PATTERN MATCH LOOKUP LOGIC
POLICING ENGINE
PACKET BUFFER
138
140
142
144
146
148
150
152
154
156
158
160
162
164
166
168
170
130
132
134
136

FIG. 4

Fig. 5

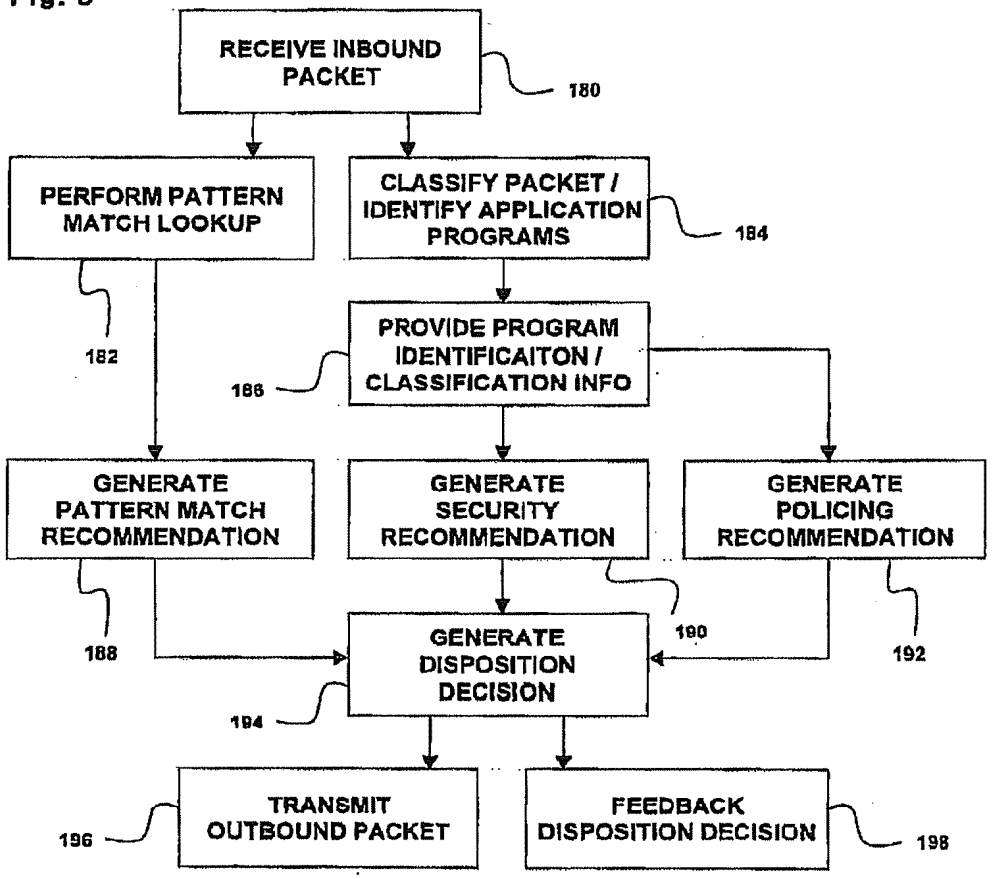
RECEIVE INBOUND PACKET
180
PERFORM PATTERN MATCH LOOKUP
CLASSIFY PACKET / IDENTIFY APPLICATION PROGRAMS
184
182
PROVIDE PROGRAM IDENTIFICAITON / CLASSIFICATION INFO
186
GENERATE PATTERN MATCH RECOMMENDATION
GENERATE SECURITY RECOMMENDATION
GENERATE POLICING RECOMMENDATION
188
190
192
GENERATE DISPOSITION DECISION
194
TRANSMIT OUTBOUND PACKET
196
FEEDBACK DISPOSITION DECISION
198

FIG. 5

Fig. 6

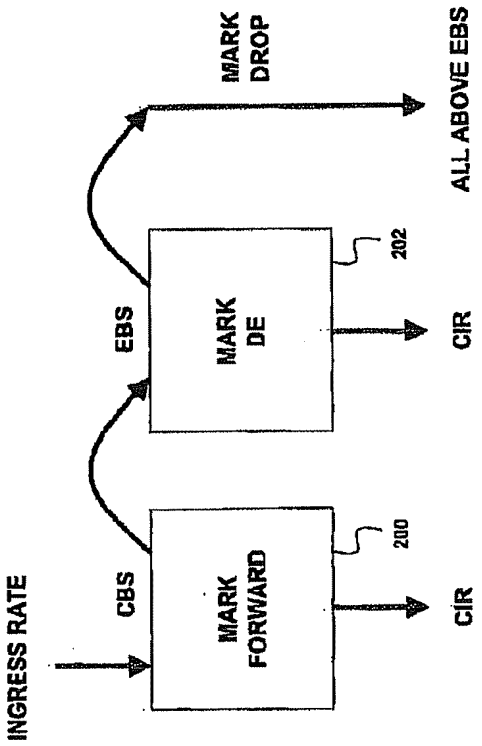
INGRESS RATE
CBS
MARK FORWARD
200
CIR
EBS
MARK DE
202
CIR
MARK DROP
ALL ABOVE EBS

FIG. 6

Fig. 7

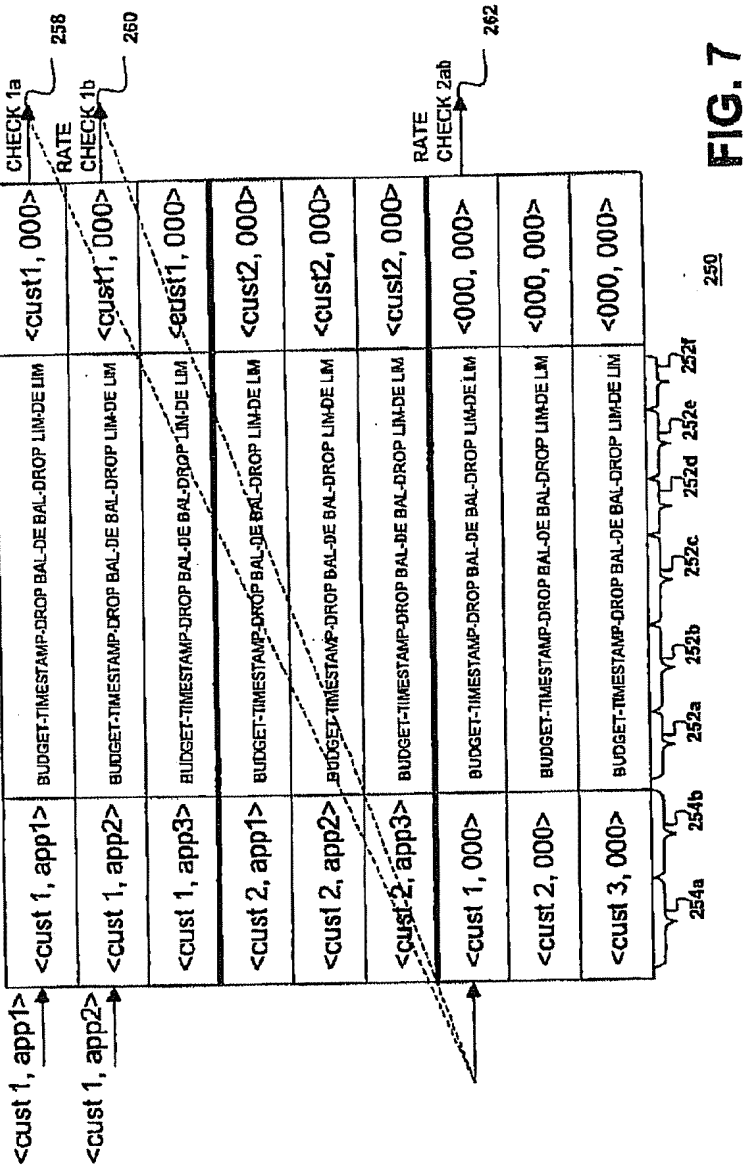
KEY
POLICING DATA
NEXT KEY
254
252
256
RATE CHECK 1a
RATE CHECK 1b
RATE CHECK 2ab
258
260
262
<cust 1, app1>
<cust 1, app2>
<cust 1, app3>
<cust 2, app1>
<cust 2, app2>
<cust 2, app3>
<cust 1, 000>
<cust 2, 000>
<cust 3, 000>
BUDGET-TIMESTAMP-DROP BAL-DE BAL-DROP LIM-DE LIM
<cust1, 000>
<cust2, 000>
<000, 000>
254a
254b
252a
252b
252c
252d
252e
252f
250

FIG. 7

Fig. 8

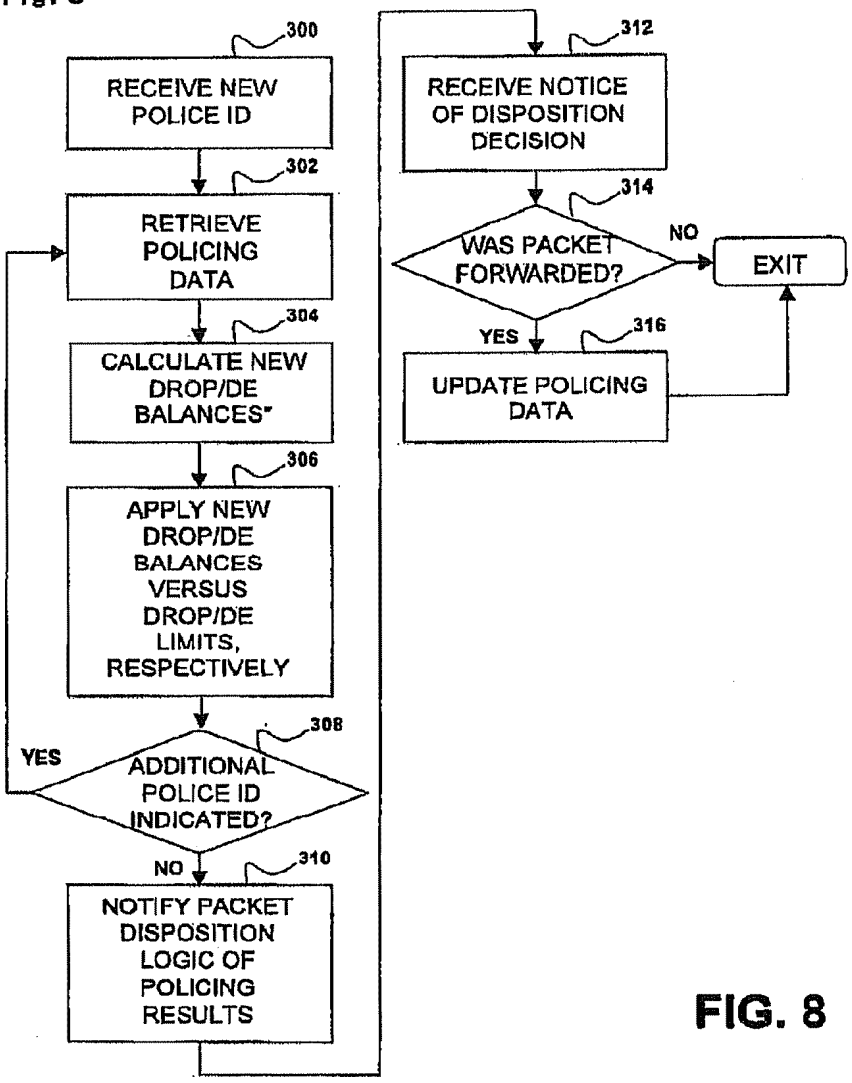

FIG. 8

* $balance_{new} = balance_{old} - [budget * (time - timestamp)] + packet\ size$

Fig. 9

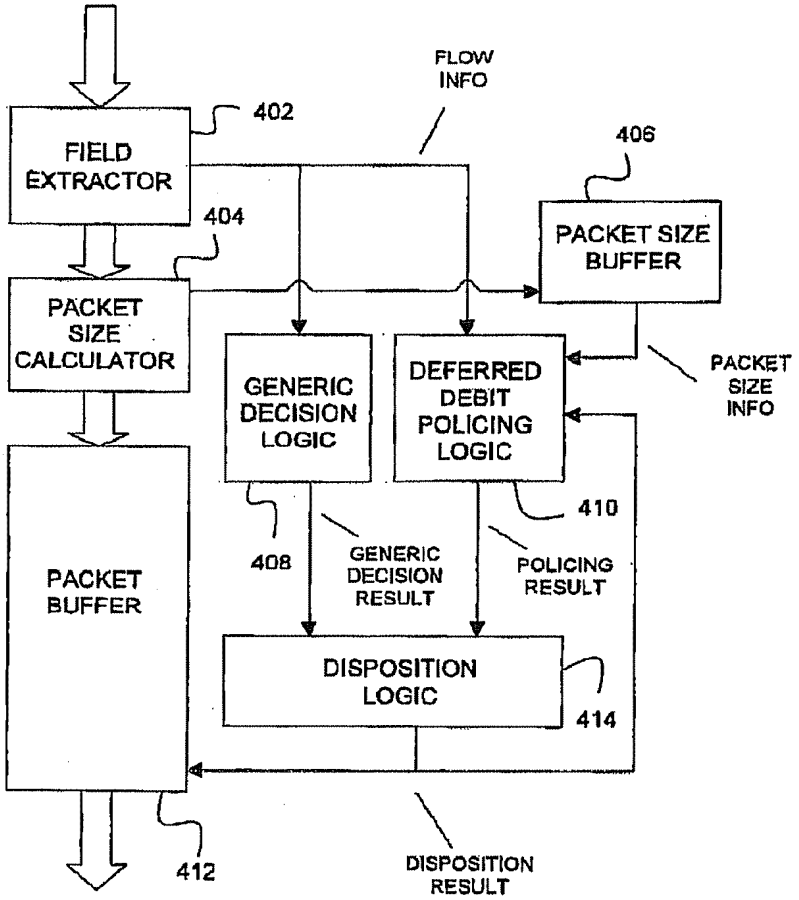
FLOW INFO
402
FIELD EXTRACTOR
406
PACKET SIZE BUFFER
404
PACKET SIZE CALCULATOR
GENERIC DECISION LOGIC
DEFERRED DEBIT POLICING LOGIC
PACKET SIZE INFO
PACKET BUFFER
410
408
GENERIC DECISION RESULT
POLICING RESULT
DISPOSITION LOGIC
414
412
DISPOSITION RESULT
400

FIG. 9

1. Abstract

A switch includes a backplane and multiple packet processors. One or more packet processors include multi-level policing logic. The packet processor receives a packet and classifies the packet into multiple policeable groups. The packet is compared against bandwidth contracts defined for the policeable groups. Nested lookups are performed for the packet in a policing database to identify the multiple groups and to retrieve policing data for the multiple policeable groups. The policing results, which may be combined into a single policing result by taking the worst case policing result, are applied to disposition logic as recommendations, and are combined with other disposition recommendations to make a disposition decision for the packet.

2. Representative Drawing

Fig. 1